APERIODIC RESPONSE TIME DISTRIBUTIONS IN QUEUES WITH DEADLINE GUARANTEES FOR PERIODIC TASKS

A THESIS
SUBMITTED TO THE FACULTY OF THE GRADUATE SCHOOL
OF THE UNIVERSITY OF MINNESOTA
BY

PAMELA ANNE BINNS

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Douglas M. Hawkins, Advisor

October 2000

Acknowledgements

Doug Hawkins was unusually broad-minded about thesis topic suitability, was patient during my early meanderings and gave sound advice including repeated reminders that a thesis is limited in scope. Max Jodeit, Bill Sudderth, and Dennis White all gave what seemed to me exceptional support and encouragement while taking courses from them. They were very helpful in providing missing or forgotten background for my new areas of study. The School of Statistics seemed accepting of my part time status.

This thesis defines models for a special case of an algorithm known as the *slack stealer* [24] developed to provide rapid response times to aperiodic tasks when scheduled in hard real-time systems. Over the last seven years, I have found opportunity to modify, extend and apply the slack stealer to multiple hard real-time systems, ranging from DARPA funded research prototypes to an FAA certified commercial avionics system. I am grateful to Honeywell Laboratories for tuition support and their flexible scheduling policies necessary for attending school while working.

With slack stealing, aperiodic performance is usually better, but no one seems to be able to systematically quantify how much better. This thesis is a step toward answering that question. In 1997, John Lehoczky (of Carnegie Mellon University) introduced me to his real-time queueing theory work which led me to look into the use of Brownian motion for modeling queueing systems. This turned out to be useful for characterizing aspects of the blocking time behavior as well as helping to develop a broader understanding of how heavy traffic approximations apply to more general queueing systems.

I am most grateful to my family for their enormous support, patience and personal sacrifice, without which I could not have completed this degree. I look forward to having more time to spend with them.

Dedication

To Steve for his extreme patience and untiring support, especially these last five years. To James for his continual forgiveness of my reduced time and attention the entire first five years of his life.

Pamela Anne Binns

334 words

Abstract

We find response time distributions for aperiodic tasks that queue for the same server with periodic tasks for which deadlines are guaranteed. The periodic task stream is a sequence of tasks with constant time between adjacent periodic arrivals and constant service times. The periodic tasks have deadlines which are times by which each task must have completed service. Deadlines are equal to the arrival time of the next periodic task. Tasks with deadlines are called hard real-time tasks.

The aperiodic task stream is a sequence of tasks with the time between arrivals drawn from an exponential distribution. The service time of each aperiodic task is also drawn from an exponential distribution. Aperiodic tasks are served in fifo order from within the aperiodic stream. The server will preempt the execution of aperiodic tasks to serve periodic tasks and guarantee that every periodic task meets its deadline.

We study two different aperiodic service disciplines called *background aperiodic service* (BGA) and *foreground aperiodic service* (FGA). BGA, also known as preemptive fixed-priority scheduling, assigns high priority to periodic tasks and low priority to aperiodic tasks. In BGA, any aperiodic tasks in service at the time of a periodic task arrival will be preempted so periodic task execution can begin immediately. FGA, a special case of the slack stealer, assigns aperiodic tasks the highest priority whenever delaying the execution of a periodic task will not result in its deadline being missed.

We develop a collection of aperiodic response time distributions. The response times are analyzed separately based on the service discipline (e.g. foreground or background). Within each service discipline, several analytic models are identified, adapted, and/or derived to characterize the response time distribution of the aperiodic

task stream. In some cases, the aperiodic system size distribution is also identified or derived. Criteria for model selection is shown to depend on specified and/or observed values of the system configuration (e.g. periodic interarrival and compute times, aperiodic interarrival and service rates, mean blocking time of aperiodics by periodics, etc.). All models and criteria are validated with simulation data.

Advisor

Douglas M. Hawkins

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Problem Description

In this thesis we study the behavior of a single queue with two classes of traffic under two different service disciplines. Each *traffic class* consists of an infinite sequence of tasks, each task arriving from some external source and each with its own service time. Within a class, the time between the arrival of the n^{th} and $(n+1)^{st}$ task is called an interarrival time. The time required to execute the n^{th} task is called a service time. Within each class, interarrival times and service times are independent of each other and drawn from common distributions.

For one class of traffic, both the service and interarrival times are constant. This traffic class is also called *periodic* or *deterministic* since tasks arrive periodically (with period H) and the next arrival time is deterministic given any previous arrival time. Each newly arriving periodic task requests C time units for its computation time.

The other traffic class is characterized by exponential interarrival and service times. This traffic class is called *aperiodic, stochastic* or *random* since the time of the next arrival is random and not periodic. The mean time between arrivals is denoted λ^{-1}. The mean service time is denoted by μ^{-1}.

A *service discipline* is a decision rule for selecting which among all queued and in-progress tasks to provide service. Only one task can receive service at any given time and the service rate is always constant (e.g. 1 sec/sec).

We impose a rule that we call the **periodic task deadline requirement**

which states that the service discipline must guarantee that all periodic tasks complete prior to the arrival of the next periodic task (the next periodic arrival is an implied deadline). Such requirements apply to *critical task functions*, where repeated failure to execute within a prescribed time limit can result in catastrophe. For example, if our periodic process were a critical feedback control process with rate H^{-1}, its output must be made available for input prior to the next iteration of the control task, otherwise the control function becomes unstable.

In most control systems, much of the traffic utilization is periodic (e.g. 60%-95%), with occasional requests from aperiodic traffic. A typical aperiodic traffic execution time (i.e. $1/\mu$) might range from 10 to 100 microseconds and periodic process periods (H values) often range from 10 milliseconds to several seconds.[1] Integrated voice and data applications is another candidate application domain, although the traffic classes will have different characteristics. Except for this mention to motive the definition and study of this problem, we will not dwell on domains of application. We also assume there is no system "benefit" in completing the execution of the periodic task prior to its deadline. The product $H\mu$ turns out to be a relevant factor in model selection. The extreme values for the ranges given are $400 = 40\text{msec}/100\mu\text{sec} = 4 \cdot 10^2$ and $100,000 = 1\text{sec}/10\mu\text{sec} = 10^5$. For critical avionics functions, a plausible range is $4 \cdot 10^2 \leq H\mu \leq 10^5$ for current day processor speeds.

We investigate this mixed stochastic and deterministic queueing model under two different but related service disciplines. In both models, the server is *preemptive-resume*. A *preemptive* server will interrupt an in-service task to start or continue service of another task. A preemptive-resume policy means that a preempted task will *resume* from exactly where it left off when it returns to the server. In other words, if a task has a service time requirement C, and is preempted at time t having

[1] Execution times will decrease in proportion with an increase in processor speed, unlike periodic rates.

completed $C(t)$ of its C required units, then upon its first return to service after time t, its remaining completion time requirement will be $C - C(t)$.

In multi-class queues, different *priorities* are sometimes assigned to different classes to give preferential treatment of one traffic class over another. In a preemptive discipline, the lower priority tasks are typically preempted to provide service to higher priority tasks. The same priority assignment rules apply to all tasks within a class. A *fixed priority assignment* assigns all tasks within a class a single fixed priority that changes with neither time nor the state of the queue. *Dynamic priority assignments* may vary as a function of time and/or system state.

In our problem of study, the periodic traffic is critical, so must have highest priority for a period of time minimally long enough to complete service prior to the next periodic task arrival. For both service disciplines, we assume that with the aperiodic stream, service is in order of arrival, or *First-In-First-Out* (FIFO). Note that periodic tasks (from a single traffic source) cannot queue by the periodic task deadline requirement.

We first investigate a service discipline called **background aperiodic service**. This service discipline falls under the category of what is known as the preemptive fixed priority server. Periodic tasks are assigned highest priority and aperiodic tasks are assigned lowest priority. If a periodic task arrives when an aperiodic task is in-service, the aperiodic task is preempted, and the periodic tasks begins execution until it completes, at which point the preempted periodic task resumes service. A task running in background has the lowest priority.

The second service discipline we investigate we call **foreground aperiodic service**. A task running in foreground has the highest priority. However, aperiodic tasks are given highest priority only when the periodic task deadline requirement is not being violated, so priority assignment is dynamic and dependent on the difference defined by the periodic task's deadline minus its remaining service time requirement.

Foreground aperiodic priority assignment rules will be made precise in Chapter 6.

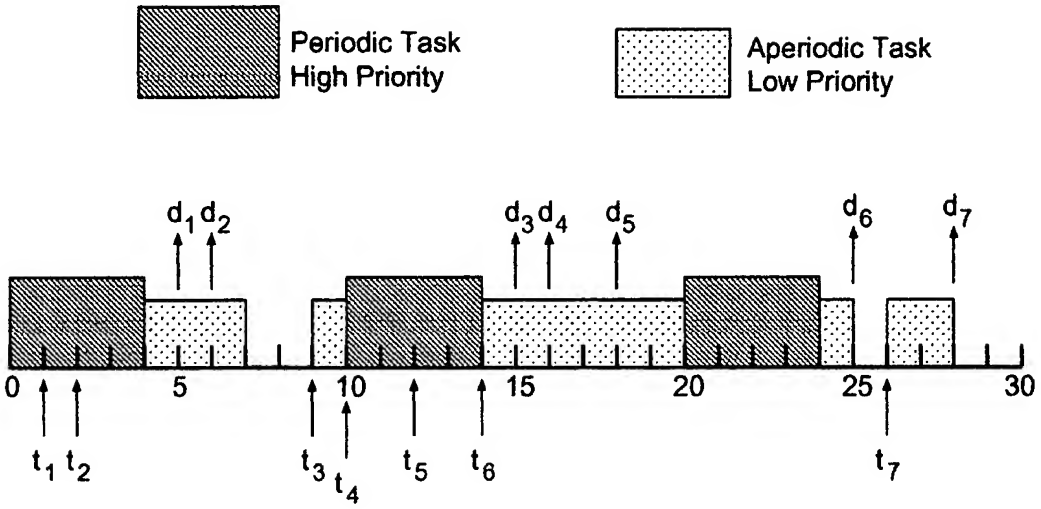
1.2 Problem Illustration

Figure 1.1 illustrates the processor timelines for background aperiodic service (top) and foreground aperiodic service (bottom) when $H = 10$, $C = 4$ and the aperiodic arrival times (t_n's) and service times (x_n's) are as shown in Table 1.1. The aperiodic interarrival times and service times are not exponential, and were chosen primarily to illustrate the definitions.

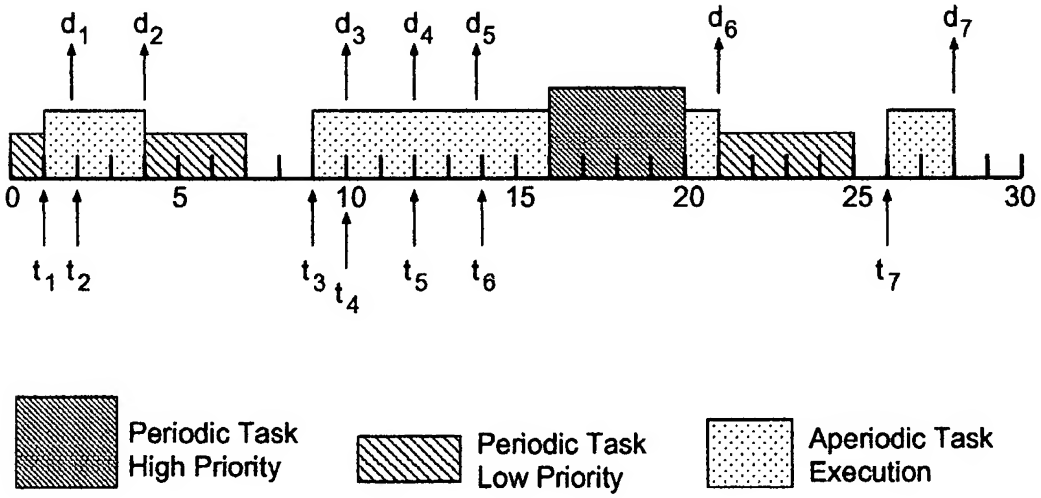
For background aperiodic service, the periodic task always has highest priority and always executes the first four units of every hyperperiod. All aperiodic tasks arriving during a periodic task's execution must wait for it to complete before the processor will spend time executing aperiodic tasks. Task priorities are completely static under background aperiodic service. The 6^{th} aperiodic task is preempted at time 20 by the (high priority) periodic arrival and resumed at time 24, when the periodic task departs.

For foreground aperiodic service, the periodic task only has the highest priority when it will otherwise miss its deadline. In Figure 1.1, aperiodic tasks always have highest priority, except when time is 16 units, since the second dispatch of the periodic task (which has received no time in the second hyperperiod), must execute four time units before time 20. At time 16, the periodic task's dynamic priority is increased (or equivalently, the aperiodic tasks' priorities are decreased), so the periodic task preempts the currently executing aperiodic task (which is the 6^{th} aperiodic task to arrive).

Also shown in Table 1.1 are the departure times for the aperiodic tasks, $d_{n,b}$ and $d_{n,f}$ when scheduled using background and foreground aperiodic service, respectively. One performance measure of interest is the aperiodic response time distribution for



Background Aperiodic Task Scheduling Timeline



Foreground Aperiodic Task Scheduling Timeline

Figure 1.1: Timeline for Example Sample Path; Top = BGA; Bottom = FGA

aperiodic task index	t_n	x_n	$d_{n,b}$	$r_{n,b}$	$d_{n,f}$	$r_{n,f}$
1	1	1	5	4	2	1
2	2	2	7	5	4	2
3	9	2	15	6	11	2
4	10	1	16	6	12	2
5	12	2	18	6	14	2
6	14	3	25	9	21	7
7	26	2	28	2	28	2

Table 1.1: Sample Path Data for Figure 1.1

both the background and foreground service disciplines. The sample background and foreground aperiodic response time values are also shown in Table 1.1 as $r_{n,b} = d_{n,b} - a_n$ and $r_{n,f} = d_{n,f} - a_n$, respectively. The next section elaborates more on measures of performance.

1.3 Thesis Objectives

Our objective is to model the *performance* of the aperiodic tasks, where modeling of physical phenomena is not equivalent to developing a mathematical anaysis of a hypothesized model. We will explore the applicability of existing models, adapt them when necessary and possible, and develop approximate models based on analytic and empirical techniques to better fit the data.

The primary performance measure of interest is the *response time* of a task. A task's response time is the time it spends in the system, or equivalently, the instant at which the task completes service (and exits the system) minus the instant at which the task arrived to the system (and either entered the queue or began service). Figure 1.2 illustrates the empirical distribution functions constructed from the background and

foreground aperiodic response times in the example of Figure 1.1. Based on this example, the response time EDF for the foreground aperiodic service discipline is better than the response time EDF for the background aperiodic service discipline.

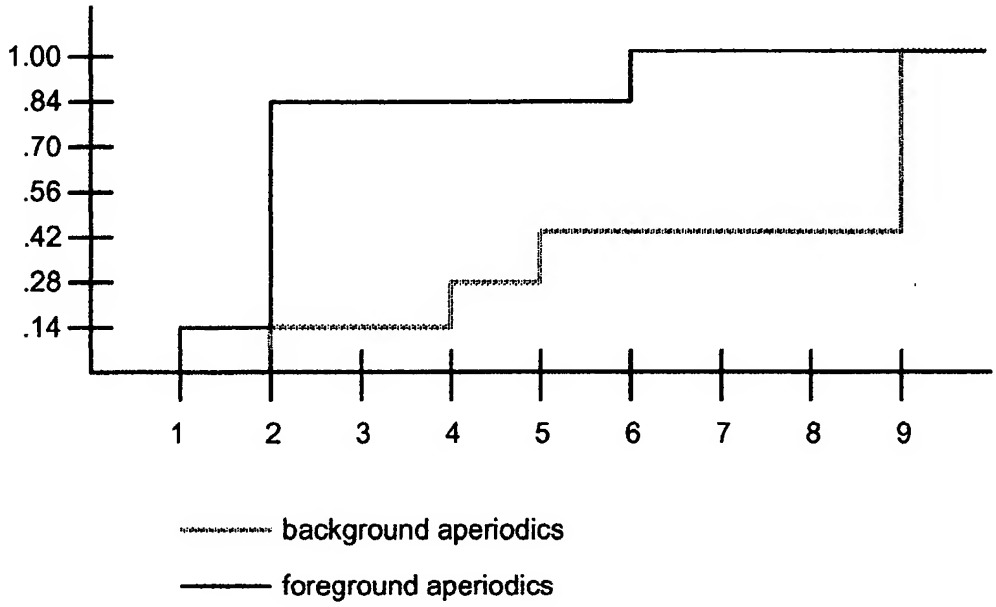


Figure 1.2: Response Time EDFs for the Example in Figure 1.1

If an objective is to produce shorter response times for a randomly observed aperiodic task, we will find the foreground aperiodic priority discipline is better. We will attempt to answer how much better, and under what conditions. We are interested in estimating the response time distribution (in contrast to say the average response time), with interest in the right tails of the distribution. We would like to be able to predict the probability that a randomly arriving aperiodic task's response time is less than some value, or perhaps compute a percentile value. In certain cases, we will also look at aperiodic queue length distributions. In all cases, we consider steady-state or time-averaged distributions (rather than transient distributions).

Even for these (relatively) simple service disciplines and traffic classes defined by our problem, a (single) precise mathematical analysis appears unachievable. In Chapters 2 and 3 we define how queueing models are transformed to stochastic processes. Analysis of the stochastic process typically takes one of several forms: (a) analytic solution for (easy) special cases and/or expected values, (b) numeric solution to state space methods, (c) analytic or numeric solution to approximating processes and (d) simulation. We restrict attention to models for describing steady state (or time-averaged) behaviour and compare our predictions to simulation results.

1.4 Thesis Organization

Chapters 2 through 4 primarily provide background and queueing theory results we will use in subsequent developments. Chapter 2 looks at results for a queue with a single server with a FIFO service discipline and one traffic stream. Chapter 3 summarizes well known results for a fixed priority preemptive resume queue using FIFO service within each traffic stream. Chapter 4 contrasts techniques introduced in Chapters 2 and 3 and establishes the context of their applicability for the analysis of our problems.

Chapters 5 and 6 comprise the development of the analyses for the response time (and system size) distributions. We did not find a single analytic model suitable for predicting response times, but rather we developed a collection of models and criteria for model selection based on the values of system parameter sets. The following categorizations for parameters were found to be useful when developing analytic methods for response time evaluation:

1. Model selection and/or parameterization depends on whether the service discipline is background (Chapter 5) or foreground (Chapter 6).

2. The total system utilization ρ ranges from light to heavy. We concentrate on conditions of heavy traffic (i.e. $\rho \approx 1$), since aperiodic response times under these conditions provide a bounding function for lighter loads. However, models for lighter total traffic are also explicitly developed under certain circumstances, and often exhibit quite different characteristics. ρ can be roughly characterized by light, moderate or heavy.

3. The periodic and aperiodic utilization, ρ_1 and ρ_2 respectively, are varied when ρ is held constant. We consider the two cases when the total traffic is primarily aperiodic and periodic. In particular, we consider $\rho_1 \in \{0.25, 0.75\}$ to represent system behavior over a spectrum of parameter ranges.

4. The hyperperiod H is the time between periodic arrivals. For simplicity, we typically assume the aperiodic service rate μ is one. The ratio of the hyperperiod to the average aperiodic service time $H/\mu^{-1} = H\mu$ ($= H$, when $\mu = 1$) turns out to be a factor when determining the suitability of various models. H is roughly characterized as short, medium and long.

With the exception of the service discipline, the factor variables are continuous valued. We provide approximate model selection criteria based on continuous values for the (model selection) factor variables.

In Chapters 5 and 6, our analyses are validated using a small set of parameters. Chapter 7 concludes with future work. Frequently used results are either cited or developed in the appendices.

Chapter 2

Some Queueing Theory Results

This chapter introduces some standard notation, typical measures of performance, and many well known results for the single server queue with a single source of traffic. Numerous books have been written on this subject (*e.g.* [1], [11], [21], [22], [31], [35], and [38]). Many texts on introductory stochastic processes with applications also contain chapters on queueing systems. (*e.g.* [20] and [32]).

The intent of this chapter is to provide just enough context to readers familiar with stochastic processes, but not with their application to queues, for the remainder of the thesis to be read stand-alone. Several results that we later use are summarized in this chapter (and Appendix A).

2.1 Queueing Definitions and Notation

For the single server queue with a single class of traffic, the following notation has become standard over the years. The queueing model described above is succinctly denoted by $A/B/1$, where A defines the interarrival time distribution and B describes the service time distribution.[1]

The *system* we consider consists of an unbounded *queue* and a single *server*. *Tasks* arrive to the system from a *traffic stream* originating outside the system. A renewal process defines the arrival times for incoming traffic. The time between

[1]More generally and commonly, it is $A/B/m/q$ where m is the number of servers (a single queue), and q is a bound on queue size. We consider only m = 1, and q = ∞, where the convention is to omit q. A convenient mnemonic for B when considering high priority traffic execution times is a denotation for blocking times.

adjacent arrivals is defined by the (common) *interarrival distribution* $A(t) = P(T \leq t)$. A sequence of interarrival times $\{T_1, T_2, T_3, ...\}$ determines the arrival time of τ_n, the n^{th} task, is $A_n = \sum_{j=1}^{n} T_j$.

Each job adds a certain amount of *work* to the system which is defined by its service time. Let $\{X_1, X_2, X_3, ...\}$ be an independent and identically distributed sample, drawn from a common service time distribution $B(x) = P(X \leq x)$. In other words, the n^{th} job to arrive to the system has service time X_n and, at the instant of arrival A_n, increases the system work by X_n. Service times and arrival times are independent of each other.

The process state variables of interest for the n^{th} task, τ_n to arrive to the system are denoted by:

$$\begin{aligned} A_n &= \text{arrival time of } \tau_n \\ T_n &= A_n - A_{n-1} = n^{th} \text{ interarrival time }, T_n \sim A \; \forall \; n \\ X_n &= \text{the service time for } \tau_n, \; X_n \sim B \; \forall \; n. \end{aligned} \tag{2.1}$$

A system with no tasks is said to be *empty* and the server will be *idle*, otherwise the server is (required to be) *busy* processing some job. These are sometimes called work conserving systems, where a system cannot remain idle if there is work to be done, nor can it produce work (at any time). It can be shown that the single server work conserving queue always has a limiting distribution whenever the average interarrival rate is less than the average service rate ([20]).

Some specific interarrival and service time distribution we use are exponential, deterministic, and gamma distributions are denoted by $\mathcal{E}(\theta)$, $\mathcal{D}(c)$, and $\mathcal{G}(\alpha, \beta)$,

respectively. To summarize distributional notation,

$$\text{For } X \sim \begin{cases} \mathcal{D}(c) & P[X = c] = 1 \text{ where c is constant} \\ \mathcal{E}(\theta) & f(x) = \theta e^{-\theta x}[x \geq 0] \\ \mathcal{G}(\alpha, \beta) & f(x) = \alpha^{\beta} x^{\beta-1} e^{-\alpha x}[x \geq 0] \\ \mathcal{P}(\lambda) & P[X = k] = \frac{\lambda^k}{k!} e^{-\lambda} \ [k \in \{0, 1, 2, ...\}] \end{cases} \tag{2.2}$$

We also use the notation $[X \in B]$[2] to denote the indicator or Bernoulli random variable taking on the value 1 if the value of X lies in a set B and zero otherwise. As is customary, we abbreviate the phrase "independent and identically distributed" with *iid.*

The *scheduling discipline* defines the decision rules the processor uses when selecting which among the currently queued jobs to process. Unless otherwise stated, we consider only the *First-In-First-Out* (FIFO) scheduling discipline, where jobs are processed to completion in the order in which they arrrive to the system. Two processes of frequent interest are the *queue length process* and the *virtual waiting time process.* The waiting time is the time a job spents in the queue before it reaches the server (for the first time). The virtual waiting time (defined at a point t) is the waiting time a job would experience if it arrived at time t.

Sometimes it is easier or more informative to consider two closely related processes, *the response or system time* and the *system size* which is the number of tasks in the system. As one might readily guess, the number in the system is zero when the server is idle, otherwise it equals the number in the queue plus one. The response time is the time spent in the queue plus the time spent in service. When a service is uninterrupted, the response time is then the convolution of the waiting

[2]Donald Knuth adopted this notation, which we find convenient when expressing integrals. For example, $\Pr(X \in B) = \int_{\Omega} [X(\omega) \in B] \, d\omega$.

time and the service time. When service times are interrupted (such as in priority queueing or in some non-FIFO service disciplines), the response time distribution is more difficult to obtain, and often has no known analytic solution.

When refering to limiting distributions, we denote response times, waiting times, queue lengths, and number in the system by R, W, Q, and N, respectively. For example, $Q(q) = P(Q \leq q)$. Q and N are non-negative discrete random variables, R is non-negative and continuous, and W is non-negative, and everywhere continuous except at 0 where $W(0) > 0$. The notation we use for derived steady state distributions, their parameters and moments are summarized in Table A.1.

It is common practice to use the same mnemonics to refer to state variables describing the process as it evolves over time. The context will (hopefully) make clear the meaning. For example, $Q(t) = Q_t$ is the number in the queue at time t. To complicate things slightly, we adopt the common notation $W(t) = W_t$ to mean the work in the system at time t. In fact, $W(t)$ is the *virtual* waiting time for a FIFO queue, which is the waiting time if an arrival were to occur at time t. We also sometimes use the notation $A(t) = A_t$ and $D(t) = D_t$ to represent the number of arrivals and departures in $[0, t]$, respectively. Again, the context should make clear the meaning. State variables are summarized in Table A.2.

Within a single task stream, we consider only non-preemptive disciplines. For a single server work conserving queue, both the number in the system and the number queued are invariant under different service disciplines. When choosing to start a new task, selecting an arbitrary task is equivalent to selecting the task at the head of the queue, since the execution times are *iid.* Thus, queue length distributions are also of interest since they can give measures of system saturation independent of the service discipline. For example, the probability the system is idle is the probability the system contains no customers. In some cases we will develop estimates for queue length distributions. For the FIFO discipline, the relationship between waiting time

and the number in the system is relatively straightforward.

2.2 The M/M/1 Queue

For the M/M/1 queueing model, both A and B are exponential, with interarrival and service rates λ and μ, respectively. The M stands for Markov. So $A(t) \sim \mathcal{E}(\lambda)$ and $B(x) \sim \mathcal{E}(\mu)$. Steady state solutions for the distributions of R, W, Q, N, and many other quantities all have analytic closed form solutions in the M/M/1 queue and can be found in many introductory texts (e.g. [20], [21], [38]). We develop just a few of the results we use, then cite and summarize others in Table B.1. Transient results require greater analytic machinery, but many results are known ([35]).

When solving for N, system state is simply the number of tasks. The steady state distribution of N is found by solving the stationary equations $\pi = \pi\mathbf{N}$, with boundary condition $\pi\mathbf{e} = \mathbf{1}$, shown in Equation 2.3.

$$\begin{array}{ll} \lambda n_k + \mu n_k = \mu n_{k+1} + \lambda n_{k-1} & \text{for } k \in \{1, 2, 3, ...\} \\ \lambda n_0 = \mu n_1 & \text{with} \\ \sum_{k=0}^{\infty} n_k = 1 & \text{as boundary conditions} \end{array} \tag{2.3}$$

Writing system utilization as $\rho = \lambda/\mu$, then solution to Equation 2.3 is easily calculated and given in Equation 2.4

$$n_k = P(N = k) = \rho^k(1 - \rho). \tag{2.4}$$

For FIFO queues, the response time distribution of the M/M/1 queue is directly caclulable once recalling that the sum of k random variables, each $\mathcal{E}(\mu)$, is $\sim \mathcal{G}(\mu, k)$

and given in Equation 2.5.

$$\begin{aligned} P(R \leq x) &= \sum_{k=1}^{\infty} P(R \leq x | N = k) n_{k-1} \\ &= \sum_{k=1}^{\infty} \int_0^x \frac{\mu^k}{(k-1)!} t^{k-1} e^{-\mu t} dt (\frac{\lambda}{\mu})^{k-1} (1 - \frac{\lambda}{\mu}) \\ &= \int_0^x \mu (1 - \frac{\lambda}{\mu}) \sum_{k=0}^{\infty} \frac{(\lambda t)^k}{k!} e^{-\mu t} dt \\ &= 1 - e^{-(\mu - \lambda)x}. \end{aligned} \tag{2.5}$$

Two things are worth noting about Equation 2.5. First, it is only valid for the FIFO service discipline.[3] Second, it is correct to condition on queue length for a newly arriving task since Poisson Arrivals See Time Averages, a fact we will find useful.[4] This property is more tersely referred to as the PASTA property, and it applies quite generally. A precise statement of the PASTA property can be found in Lemma C.2.1.

It is often possible to explicitly calculate first moments of first passage times for the M/M/1 queue. One value of interest is the mean time to first transition from state $k+1$ to state k, for $k \in \{0, 1, 2, ...\}$, which we denote by $E[k+1 \rightarrow k]$. It can be shown [19] that

$$E[k+1 \rightarrow k] = (\mu - \lambda)^{-1} \text{ for } k \in \{0, 1, 2, ...\}. \tag{2.6}$$

An application of Equation 2.6 is the calculation of the expected time of a busy interval. Suppose a task arrives at time t to an empty queue, at which point the process state is one. The busy interval ends when process state first reaches zero after time t. Equation 2.6 tells us that $E[1 \rightarrow 0] = (\mu - \lambda)^{-1}$. It is interesting to note that the mean response time is equal to the mean time of a busy interval. The Markov property of the M/M/1 queue gives us a generalization to Equation 2.6 shown

[3] In fact, for non-FIFO service disciplines, the response time distributions are generally not known for even for the M/M/1 queue.

[4] For example, the PASTA property implies that the virtual waiting time is the waiting time for any Poisson arrival process.

in Equation 2.7.

$$E[k+m \rightarrow k] = m(\mu - \lambda)^{-1} \text{ for } k \in \{0,1,2,....\} \text{ and } m \in \{1,2,3,...\}. \tag{2.7}$$

Much more is known about the M/M/1 queue, and more generally about continuous-time birth-death Markov processes. For example, the distributions of the queue length, the number served in a busy interval, and of the busy and idle period durations are also known. Results we use are summarized in Table B.1 for easy reference.

2.3 The M/G/1 Queue

In the $M/G/1$ queue, the arrival process is Poisson, but the service times can be general. G stands for general (i.e. G could be any of $\mathcal{E}$, $\mathcal{G}$, $\mathcal{D}$ or other distribution). For the $M/G/1$ queue, the interarrival time distribution is $\mathcal{E}(\lambda)$ and the service time distribution B has mean m_b and variance $\sigma_b^2 < \infty$. The results cited below are well known and can be found in most texts (e.g. [20],[21], [11]).

The $M/G/1$ queue is of interest to our problem, since a priority queue can often be approximated by reducing it to an $M/G/1$ queue using the concept of a *completion time*. A completion time for a task τ is measured as the time from the first instance τ receives service to the time at which τ completes. Completion times and their use are described more in Section 3.2.

The M/G/1 queue is a semi-Markov process, containing an imbedded Markov Chain for the number of customers in the system defined at task departure instants. This is the result of Lemma 2.3.1

Lemma 2.3.1 (An M/G/1 Imbedded Markov Chain.) *Since the service time*

distribution is not exponential, transition probabilities will depend on the time a task has been in service. So, any process defining system state must contain information equivalent to time since start of service. For example, $[N(s), X_0(s)]$, *where* $X_0(s) =$ *the time current task has been in service, is a continuous state Markov process (with sample paths that jump).*

Consider the set of points at which departures occur, say $s_1, s_2, s_3, \ldots$. *At these points,* $X_0(s_n) \equiv 0$, *making* $\{N(s_n)\}$ *a sample realization of a Markov chain.*

Let v_k = the number of arrivals to the system during the execution of task τ_k.[5] The defining equations for the number in system in the imbedded Markov chain developed in Lemma 2.3.1 is given by Equation 2.8

$$N_{k+1} = \begin{cases} N_k + v_{k+1} - 1 & \text{for } N_k > 0 \\ v_{k+1} & \text{otherwise} \end{cases} \tag{2.8}$$

Using the relationship in Equation 2.8 and moment generating functions, results for the M/G/1 queue are in terms of the service time LaPlace transform, $\hat{B}(s) = E[e^{-sX}]$, where X is the service time random variable with distribution B. The LaPlace transform of the number in the system is given by

$$\hat{N}(z) = \hat{B}(\lambda(1-z))\frac{(1-\rho)(1-z)}{\hat{B}(\lambda(1-z)) - z}. \tag{2.9}$$

The development of Equation 2.9 was based only on departure instants. However, another important result holds for the M/G/1 queue which we highlight in Lemma 2.3.2.

[5]The density of v_k is simply $P(v_k = m) = \int_0^\infty \frac{(\lambda x)^m}{m!} e^{-\lambda x} dB(x)$, for $m \in \{0, 1, 2, \ldots\}$ and $k \in \{1, 2, 3, \ldots\}$.

Lemma 2.3.2 (Departures See Time Averages in the M/G/1 Queue) *In the limit, the number of times the imbedded Markov departure time process transitions from k to $k+1$ differs by no more than one from the number of transitions from $k+1$ to k. So given the PASTA property for arrivals, one might expect it to also hold for departures. For a simple but more formal and convincing argument, see [11] (pp. 221).*

Application of Lemma 2.3.2 renders Equation 2.9 valid for all time points. Inverting Equation 2.9 remains. This is sometimes possible by inspection (e.g. $\hat{N}$ for the M/D/1). In theory, numeric inversion is an option when the service time distribution has a density with respect to Lebesgue measure.[6] However in practice, only moments are usually sought.

Under the assumption of FIFO service, the LaPlace transform for the response time (which is the product of the transforms for the waiting time and service time) can also be found and is represented in Equation 2.10

$$\hat{R}(s) = \hat{B}(s)\frac{s(1-\rho)}{s-\lambda+\lambda\hat{B}(s)}. \tag{2.10}$$

Much more is known about the $M/G/1$ in steady state. Its transient behavior is treated in detail in [35].

2.4 Numeric Methods

Most practical queueing problems are not described by stochastic processes that submit nicely to analytic solutions. Instead, they are solved using one or more of a variety of numeric methods. Beyond this section, we won't focus on numeric solution

[6] Of course, a density can be approximated to an arbitrary degree of precision by using very steep slopes about points of discontinuity.

methods since their solution rarely provides insight into the nature of the problem, and inferences can rarely be made beyond the particular parameter settings for which the problem was solved.

We will identify a handful of properties that can greatly complicate the prospects of analytic solution. We will later see that our queueing models exhibit several of these complicating properties. We also mention a few of the more common solution techniques, some of which were used in Section 2.3.

When a queueing model is non-Markovian, efforts to somehow *convert* it to a Markov process for which known results apply are often tried. Among these conversion techniques are approximation by diffusion processes, the method of supplementary variables, the method of imbedded Markov chains and the method of stages.

Examples of using diffusion approximations for queue length and virtual waiting time processes for the G/G/1 queue (near saturation i.e. $\rho = \lambda(\mu)^{-1} \approx 1$) are developed in Section 2.5. In Section 2.3, the number in the system process, $N(t)$ was first converted to a continuous state Markov process by adding a new state variable in the process state description. Now $\tilde{N}(t) = [N(t), X_0(t)]$ represents the (system size) process state at time t. Increasing the dimensionality of the state vector with additional information so the relevant past information is contained in the current state is known as the *method of supplementary variables.*

$[N(t), X_0(t)]$ is a continuous-state Markov process (when $B(x)$ is absolutely continuous), and solving continuous-state Markov processes is still a formidable task. The *method of imbedded Markov chains* involves looking at the process only at points defining an imbedded Markov chain. In the M/G/1 queue, we saw that departure instants formed one set of time points defining an imbedded Markov chain. Similarly, for the G/G/1 queue, the start of idle intervals defines a set of imbedded points ([36]). The behavior at these imbedded chains may, but typically will not, be the same as overall system behavior.

Sometimes states are introduced to approximate a system of interest. For example, a constant random variable $X \sim \mathcal{D}(D)$ can be approximated by a gamma random variable $G \sim \mathcal{G}(\alpha = k, \beta = D(k\mu))$, when letting $k \to \infty$. To model a deterministic execution time, a task's execution consists of k sequential identical phases, where each phase is exponentially distributed with rate μ. As k increases, the approximation improves. This approximation example illustrates what is known as the *method of stages*. The state space remains discrete, but the state description now contains a new state variable (i.e. a supplementary variable) describing the stage of execution. The original continuous state process has been approximated by a (potentially large) discrete state Markov process.

Using discrete state methods to approximate an inherently continuous phenomena (such as response times) is likely to result in a very large state space, where numeric solution is required. There are a number of available numeric solution techniques (but numeric instability is not uncommon) for solving (possibly very large) sets of linear equations, which can be used for finding the stationary distribution of the Markov chain. Assessing the quality of the approximation (to the original model) is yet another problem that also need not lend itself to analysis.

An alternative approach is to use a continuous state process to define the queueing model. All the queueing processes we have considered so far have renewal processes that form the input flows, and when the server is busy for extended periods of time, the output flows. The interarrival and service time distributions have depended on neither time nor system state. When we consider the background and foreground priority server models as a reduction to an M/G/1 queue, the completion times will depend on both time of arrival and on system state.

When the process is non-Markovian (which it is in our problem), there are no general solutions for performance distributions. The best we can hope for is special case analyses or approximation by continuous state Markov processes, which

constitutes much of the material in Section 2.5.

2.5 The G/G/1 Queue

For the G/G/1 queue, both the interarrival time and service time distributions are arbitrary. When the service time distribution is exponential, the model is a G/M/1 queue for which many analytic results have been found. However, when using completion times to approximate a preemptive priority queue, the distribution for the completion time is not approximately exponential, so the many known results for G/M/1 model are not useful to us. Without further restrictions on either the interarrival time or service time distributions, not even first moments of performance measures need have (known) closed form solutions.

2.5.1 G/G/1 Process Formulation

A direct and commonly presented representation for the G/G/1 queue is Lindley's integral equation (also known as a Weiner-Hopf type equation). An analysis of Lindley's integral equation requires analytic machinery well beyond the scope of this (student and her) thesis. Nonetheless, a sketch of the development of Lindley's integral equation proves worthwhile since it forms the basis for a compact representation of the queueing models we will study when the hyperperiods are intermediate in length. The development presented here largely parallels that given in [21].

The development of Lindley's equation applies to waiting times, the time a task spends in the queue before starting service. In addition to the notation in

Equation 2.1, we make use of the following defining relationships:

$$\begin{aligned} U_n &= X_n - T_{n+1} = \text{ system change in work by } \tau_n \text{ relative to } A_{n+1} \\ W_n &= \text{ time } \tau_n \text{ spent waiting before starting service} \\ C_n(u) &= \Pr[U_n \leq u] \\ W_n(y) &= \Pr[W_n \leq y] \\ W_{n+1} &= \max[0, W_n + U_n] = (W_n + U_n)^+ \end{aligned} \tag{2.11}$$

It is the last relationship in Equation 2.11 that we use as a basis for our process simulation model. Here is one possible approach to numerically solving for the limiting waiting time distribution. By definition,

$$C_n(u) = \int_{t=0}^{\infty} P[X_n \leq u + t | T_{n+1} = t] dA(t).$$

Noting that X_n is independent of T_{n+1}, it follows that

$$C_n(u) = C(u) = \int_{t=0}^{\infty} B(u + t) dA(t).$$

Also, for $y \geq 0$ and $W_{n+1}(0^-) = 0$,

$$\begin{aligned}
W_{n+1}(y) &= \int_{0^-}^{\infty} P[U_n \leq y - x | W_n = x] dW_n(x) \\
&\quad \text{by definition of } W_{n+1} \\
&= \int_{0^-}^{\infty} C_n(y - x) dW_n(x) \\
&\quad \text{by independence of } U_n \text{ and } W_n, \text{ so} \\
W(y) &= \int_{0^-}^{\infty} C(y - x) dW(x) \\
&\quad \text{by Equation 2.11 and def of } \lim_{n\to\infty} W_n \\
&= -\int_{0^-}^{\infty} W(x) dC(y - x) \\
&\quad \text{by integration by parts} \\
&= \int_{-\infty}^{y} W(y - x) dC(x) \\
&\quad \text{by a change of variables.}
\end{aligned} \tag{2.12}$$

The three relationships defining $W(y)$ in Equation 2.12 are different forms of Lindley's integral equation, a type of Weiner-Hopf equation. Not even an explicit expression for $\bar{W}$, the mean wait is known without making some assumptions about the general service and interarrival distributions.

It may be worth noting the PASTA property need not hold for the G/G/1 queue. For example, consider the D/D/1 queue where the number of tasks an arrival sees in the system (not counting the arriving task) is zero with probability one. If we were to pick a random time and then observe the number of tasks in the D/D/1 queueing system, we would see one task with probability C/H and zero tasks with probability $1 - C/H$ (which is the distribution an arrival would see if the PASTA property were applicable).

As an alternative to an exact analysis, the queueing process can sometimes be well approximated by a simpler process, for which analytic solution is possible. The remainder of this chapter discusses diffusion approximations for queue length

and waiting time distributions.

2.5.2 G/G/1 Heavy Traffic Approximations

In certain cases, diffusion approximations have been shown to provide reasonable approximation of queueing models under *heavy traffic* conditions. Under heavy traffic conditions, the system is near saturation (i.e. $\rho = \lambda/\mu \approx 1$). Figure 2.1 is an illustration of the plausibility of this approximation, which shows the queue length process of an M/M/1 queue as it evolves over time. Note that the horizontal time axis, with $t \in [0, \infty)$, acts as a reflecting barrier since queue lengths never go negative.

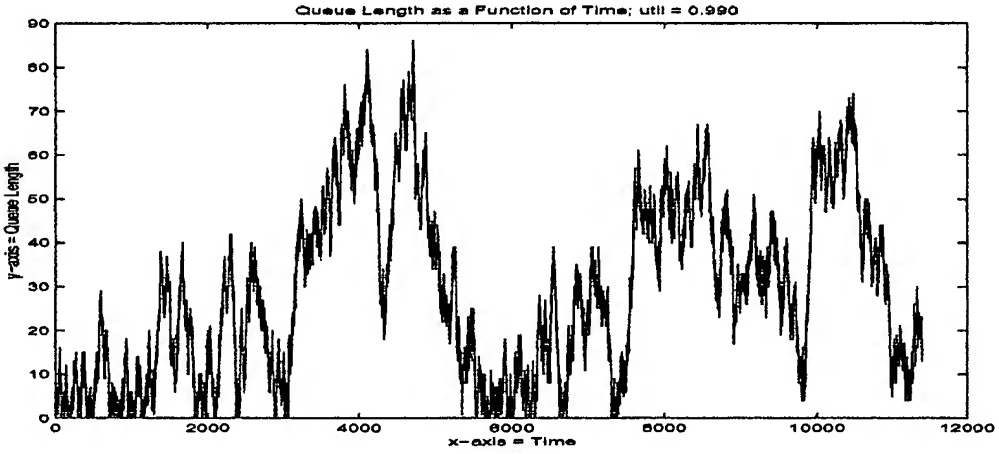


Figure 2.1: Queue Length Sample Path in Heavy Traffic

Steady state solutions for approximating the G/G/1 (and much more general queueing structures) have been found when the diffusion coefficients are constant (i.e. independent of state and time).

A common strategy for studying the G/G/1 queue in conditions of heavy traffic is to appropriately scale time ($t \to [t/n]$) and space (e.g. queue size, $q \to q(n^{-1/2})$) in the original queueing system, creating a sequence of random functions that converge weakly to a Brownian motion. Donsker's theorem is included in Section C.4 and

provides the underlying mathematical machinery needed to obtain the weak convergence results. To obtain the equilibrium distribution (when one exists), the limiting diffusion process needs to be evaluated as $t \to \infty$.

Whitt ([37]) appears to have been the first to find diffusion approximations using weak convergence limit theorems for a preemptive fixed priority queue. Burman ([5]) used the method of supplementary variables to find a diffusion approximation for the G/G/1 queue length process. Harrison ([13]) presents a comprehensive treatment for finding solutions to networks of queues with multiple traffic classes (but none have priorities), and an arbitrary routing structure. Harrison ([14]) also presents an analysis for the transient solution of a Brownian motion with a reflecting barrier at the origin, which has applicability to modeling transient heavy traffic behavior of the G/G/1 queue.

Solutions for diffusion processes with time and state dependent coefficients requires solution techniques for stochastic partial differential equations, which sometimes can be solved using the Ito calculus ([2]), but typically can only be numerically estimated.[7] These solution techniques are well beyond the scope of this (student and her) thesis. The literature on the subject is enormous but results for applications of even modest complexity (such as those in this thesis) are mathematically inaccessible to most and appear largely unconsolidated when they exist.

Later our simulation studies will reveal that the preemptive fixed priority diffusion approximation is often excessively optimistic when the mean aperiodic and periodic service times are not of the same order of magnitude. However, we will find cases where a diffusion process is a reasonable estimator of blocking times, so it is instructive to present the results, and sketch some of the development.

[7] The Ito calculus is also applicable for stochastic differential equations with constant coefficients.

2.5.3 Approximating the System Size Process

Define processes

$$\begin{aligned} A_t &= \text{number of arrivals in } [0,t] \\ T_t &= \text{amount of time spent busy in } [0,t] \\ \tilde{D}_t &= \text{number of departures from an ever busy server in } [0,T_t] \text{ and} \\ D_t &= \text{number of departures in } [0,t]. \end{aligned} \tag{2.13}$$

$A_t, \tilde{D}_t$, and D_t are counting processes. T_t has been called an *allocation* process and it defines the amount of time allocated to serving processes. By definition, $N_t = A_t - D_t$. We call N_t the system size process to distinguish it from the queue length process, Q_t. Some authors refer to both N_t and Q_t as the queue length process, providing context as a means for differentiation.

We now compute the coefficients for the limiting diffusion process. The asymptotic means and variances of A_t and $\tilde{D}_t$ are easily computed, since they are counting processes of renewal processes. Specifically, as $t \to \infty$, $E[A_t](t)^{-1} = \lambda$ and $E[\tilde{D}_t](T_t)^{-1} = \mu$. An application of the CLT for renewal processes (see Lemma C.1.1) gives $\text{Var}(A_t)(t)^{-1} = \sigma_a^2\lambda^3$ and $\text{Var}(\tilde{D}_t)(T_t)^{-1} = \sigma_b^2\mu^3$.

Under conditions of heavy traffic, the server is usually busy, so $T_t \approx t$ and $\tilde{D}_t \approx D_t$. Analogously, we approximate N_t by $A_t - D_t$. Further, since some queueing usually occurs, the process D_t is largely independent of the process A_t. This is in contrast to a lightly loaded server, where the departure process is highly dependent on the arrival process.

Assuming independence of A_t and D_t, the diffusion coefficients for process N_t

are now defined by a simple calculation,

$$\begin{aligned} m_N = &\ E[N_t](t)^{-1} = &\ (\lambda - \mu), \text{ and} \\ \sigma_N^2 = &\ \text{Var}(N_t)(t)^{-1} = &\ (\text{Var}(A_t) + \text{Var}(D_t))(t)^{-1} = \ (\sigma_a^2\lambda^3 + \sigma_b^2\mu^3). \end{aligned} \tag{2.14}$$

The reader unfamiliar with weak convergence arguments is encouraged to first review Section C.4, which also provides some notation. It turns out that ([2],[20],[22]) in the limiting process, the transitions are defined by the Kolmogorov forward diffusion equations (also called the Fokker-Planck equations) in Equation 2.15.

$$\frac{\partial p(x,t|x_0)}{\partial t} = -\frac{\partial}{\partial x}(m(x,t)\cdot p(x,t|x_0)) + \frac{1}{2}\frac{\partial^2}{\partial x^2}(\sigma^2(x,t)\cdot p(x,t|x_0)) \tag{2.15}$$

Fortunately, Equation 2.15 has been solved under a number of boundary conditions when $m(x,t) = m = m_N$ and $\sigma^2(x,t) = \sigma^2 = \sigma_N^2$. When $\lambda \geq \mu$ the queue grows without bound, and no proper steady state queue length distribution exists. However, for $\lambda > \mu$, we might want to consider the queue length after some long period of time. In this case, we have

$$P(N_t^\infty \leq x) = \frac{1}{\sqrt{2\pi\sigma^2 t}}\int_{-\infty}^{x} e^{-\frac{1}{2}(y-mt)^2/(\sigma^2 t)}dy = \mathcal{N}(x; mt, \sigma^2 t). \tag{2.16}$$

We will find application for Equation 2.16 for describing aperiodic queue length at the end of a blocking time under the aperiodic background priority server discipline. The diffusion coefficients will, of course be different.

When $\lambda < \mu$, the boundary condition $p(x,t|x_0) = 0$ for $x \leq 0$, $t \geq 0$, and $x_0 \geq 0$ corresponds to a reflecting barrier at the origin (queue lengths and waiting times are never negative and reflect when reaching the origin). Under these conditions,

the solution to Equation 2.15 is given by

$$P(N^{\infty} \leq x) = 1 - e^{2mx/\sigma^2} \text{ for } x \geq 0 \text{ so } N^{\infty} \sim \mathcal{E}(2m\sigma^{-2}). \tag{2.17}$$

In the special case of the M/M/1 queue, $m = (\lambda - \mu)$ and $\sigma^2 = (\lambda + \mu)$. For $\lambda < \mu$, Equation 2.17 applies giving

$$\Pr(N \leq k) = 1 - e^{-2(\mu-\lambda)(\mu+\lambda)^{-1}k} \text{ for } k \geq 0. \tag{2.18}$$

Since we are approximating a discrete random variable by a continuous one, Equation 2.18 will differ from an exact solution, which for the M/M/1 queue is

$$\Pr(N \leq k) = 1 - \rho^{k+1} = 1 - \rho\rho^{k}. \tag{2.19}$$

When $\rho \approx 1$, it suffices to show that $e^{-2(\mu-\lambda)(\mu+\lambda)^{-1}} \approx \rho$. To see that the expression in Equations 2.18 reasonably approximates the value in Equation 2.19, take $\mu = 1$ and the first two terms in the Taylor's expansion of $e^{-2(1-\lambda)(1+\lambda)^{-1}}$, giving $1 - 2(1 - \lambda)(1 + \lambda)^{-1} \approx \lambda$, or $(1 - \lambda)^2 \approx 0$ which follows from the assumption $\lambda \approx \mu = 1$.

2.5.4 Approximating the Virtual Waiting Time Process

To calculate the virtual waiting time of a FIFO queue, we introduce several new process definitions using variables and processes previously defined in Equations 2.1

and 2.13. Let

$$\begin{aligned} X_t = x_1 + x_2 + ... + x_{A_t} &= \text{cumulative work requested in } [0,t] \\ Y_t = X_t - t & \\ I_t = -\inf_{0\le s\le t} Y_s &= \text{cumulative idle time in } [0,t] \\ T_t = t - I_t &= \text{cumulative busy time in } [0,t] \text{ and} \\ W_t = X_t - T_t &= \text{pending work at time t.} \end{aligned} \tag{2.20}$$

The process definitions in Equation 2.20 apply in much greater generality than we have considered. Assume the queue begins empty, so $W_0 = 0$ and the number of departures cannot exceed the number of arrivals, in which case $W_t = x_{D_t+1} + x_{D_t+2} + ... + x_{A_t}$. Letting $t \to \infty$

$$\begin{aligned} m_W &= \lim_{t\to\infty} E[W_t](t)^{-1} &&= (E[\textstyle\sum_{k=D_t+1}^{A_t} x_k)](t)^{-1} \\ &= (E[D_t+1] - E[A_t])(\mu t)^{-1} &&= (1-\rho) + \lim_{t\to\infty}(\mu t)^{-1} \\ &= (1-\rho), \end{aligned} \tag{2.21}$$

where the second line in Equation 2.21 follows from Wald's identity and $\rho = \lambda/\mu$.

Now for $n \in \{1,2,3,...\}$ define sequences of processes:

$$\begin{aligned} W_n(t) &= [W(nt) - (\rho-1)nt]n^{-1/2}, \\ A_n(t) &= [A(nt) - \lambda nt]n^{-1/2} \text{ and} \\ S_n(t) &= [\textstyle\sum_{j=1}^{[nt]}(x_j - \mu^{-1})]n^{-1/2}. \end{aligned} \tag{2.22}$$

The following weak convergence results have been shown ([37]).

$$\begin{aligned} S_n &\Rightarrow S \text{ where } S \sim \sigma_b \xi^1 \\ A_n &\Rightarrow A \text{ where } A \sim (\lambda^3 \sigma_a^2)^{-1/2} \xi^2, \text{ and} \\ W_n &\Rightarrow \lambda^{1/2} S + \mu^{-1} A, \end{aligned} \tag{2.23}$$

where ξ^1 and ξ^2 are independent Wiener processes. (See Section C.3.) From Equation 2.23, the variance is easily calculated. $\mathrm{Var}(W_n) = \lambda \mathrm{Var}S + \mu^{-2}\mathrm{Var}A = \lambda\sigma_b^2 + \mu^{-2}\lambda^3\sigma_a^2$. Summarizing, the diffusion coefficients for the waiting time are

$$m_W = (\rho - 1) \text{ and } \sigma_W^2 = \lambda\sigma_b^2 + \mu^{-2}\lambda^3\sigma_a^2. \tag{2.24}$$

For $m_W < 0$, the limiting distribution of W^∞ is found using Equation 2.17 with coefficients given in Equation 2.24. Equation 2.25 gives the general case,

$$\Pr(W^\infty \leq t) = 1 - e^{-2(1-\rho)(\lambda\sigma_b^2 + \mu^{-2}\lambda^3\sigma_a^2)^{-1}t} \text{ for } t \geq 0. \tag{2.25}$$

For the special case of the M/M/1 queue, the limiting virtual waiting time distribution becomes Equation 2.26.

$$\Pr(W^\infty \leq t) = 1 - e^{-(1-\rho)\mu^2\lambda^{-1}t} \text{ for } t \geq 0. \tag{2.26}$$

The exact waiting time distribution for the M/M/1 queue is given by $1 - \rho e^{-(\mu-\lambda)x}$. Under conditions of heavy traffic, $\rho \approx 1$, so $1 - \rho e^{-(\mu-\lambda)x} \approx 1 - e^{-(\mu-\lambda)(\mu/\lambda)x}$, which is Equation 2.26.

Chapter 3

Priority Queues

In this chapter we introduce notation and modeling constructs helpful in the analysis of priority queues. We also define typical measures of performance for priority queues, and give several known results for the single server queue with multiple sources of traffic. Results that we use later are included in this chapter for future reference.

3.1 Performance Measures and Notation

Our system now consists of a single server with r traffic *classes*, each originating from a distinct traffic stream outside the system. In our case, $r = 2$, but many of the concepts generalize. In addition to the server, there is queueing. One can think of either a single queue that is sorted by priority, or of r distinct queues where jobs of class j queue in the j^{th} queue. We have tried to consistently denote periodic subscripts by 1 and aperiodic subscripts by 2.

Definitions for parameters and the limiting performance distributions introduced here are summarized in Table A.1. The notation conventions introduced in Chapter 2.1 are extended to include a subscript for the traffic class. For example, R_c would refer to the steady state response time distribution for class c tasks. For X a random variable, or F a distribution function, we use the notation $X \sim Y$ and $Y \sim F$, when $X = Y$ a.s. or when Y has distribution F, respectively.

Each job class $j \in \{1, ..., r\}$ has an interarrival time distribution, $A_j(x) = \Pr(T_{j,n} \leq x)$, $n \in \{2, 3, ...\}$ where $T_{j,n}$ is the time between the $(n-1)^{st}$ arrival of class j and the n^{th} arrival of class j. The arrival process for each class then forms

a renewal process. Since the only classes of interarrival processes we consider are either exponential, or deterministic (starting the system with a deterministic arrival at zero), we may assume that the first interarrival time distribution is equal to those of all subsequent interarrival times. Since we are interested in equilibrium distributions, it turns out that we may make the same assumption for more general arrival processes. All arrival processes are assumed to be independent.

The mean time between $T_{j,n}$ and $T_{j,n+1}$ is denoted by λ_j^{-1}. λ_j is called the arrival rate for class j. A counting process $N_{A,j}(t)$, counting the number of class j arrivals in $(0,t)$ is used when defining the queue length and other processes.

Recall (from the Introduction) that the *scheduling discipline* defines the decision rules the processor uses when selecting which among the currently queued jobs to process. For *priority* queues, when a departure occurs, the processor will always select the job with the highest *priority* to start processing next. A *preemption* occurs when the processor is serving a class j job class and a job with higher priority arrives or otherwise becomes available. Note that preemptive schedulers need not provide uninterrupted service to jobs. Job priorities can be either *fixed* or *dynamic*. A fixed priority does not change with time, whereas a dynamic priority does. The *background* and *foreground aperiodic service* disciplines are examples of fixed and dynamic priority scheduling disciplines, respectively.

Similarly, each job class $j \in \{1, ..., r\}$ has a service time distribution $B_j(x) = \Pr(X_{j,n} \leq x)$, $n \in \{1, 2, ...\}$. The service time distributions are also assumed to be independent of each other and of the interarrival processes. The mean service time (in the absence of interruption) for class j is defined by m_j. The mean service rate (in the absense of interruptions) for class j is $\mu_j = m_j^{-1}$. An analogous counting process for completed services $N_{D,j}(t)$ is defined by the number of class j departures (or completed class j services that have occurred in $(0,t)$).

Unless otherwise stated, we consider only the *First-In-First-Out* (FIFO) scheduling discipline within each job class. In other words, a class j job, once started will be completed before any other class j job is started. And they will be started in the same order in which they arrived.

The notation used to describe the process' state variables is summarized in Table A.2. Again note that many of the same symbols (with perhaps different subscipts) are used to define limiting distributions as state variables. The processes of typical interest are once again the *queue length process* (or system size) extended in the natural way to $Q(t) = (Q_1(t), Q_2(t), ..., Q_r(t))$, and the *virtual workload process* $W(t) = (W_1(t), W_2(t), ..., W_r(t))$. An unsubscripted variable might also denote a system variable. For example, $Q(t) = Q_1(t) + Q_2(t)$ is the total number of tasks queued at time t.

When priorities are fixed, all the results in Chapter 2 apply to jobs of class 1 (i.e. jobs with the highest priority), in which case interest is focused on the lower priority classes. For jobs of class $k > 1$, the class k virtual workload differs from the class k virtual waiting time, since in a preemptive priority discipline all jobs of class $j < k$ will receive service prior to providing service to any jobs of class k. Also, the execution of class k jobs will be interrupted by any new arrivals of class $j < k$. In other words, W_k is the virtual waiting time a class k job would see only if $W_j = 0$ for all $j < k$ and there are no new arrivals for any class $j < k$ while the queue contains class k jobs.

3.2 Completion Times

When service within each stream is FIFO, the concept of a *completion time* can be used to (approximately) transform the analysis of a preemptive priority queue to an analysis of a queue with only a single traffic class. When the arrivals are Poisson in

the stream of interest, the priority queue can be studied as an M/G/1 queue. The completion time of a task is defined to be the time between the first and last instants of execution. In other words, the start of the completion time begins when a task first transitions from waiting to service. In a preemptive priority queue, tasks can be preempted by higher priority tasks. The execution time of these higher priority tasks contributes to the completion time of the task being preempted. The completion time ends when the task exists the system.

Equation 3.1 shows the probability of task type, when considering a task selected at random.

$$\begin{aligned} \text{P(periodic arrival)} &= \tfrac{\lambda_1}{\lambda_1+\lambda_2} \text{ and} \\ \text{P(aperiodic arrival)} &= \tfrac{\lambda_2}{\lambda_1+\lambda_2} \end{aligned} \tag{3.1}$$

One case of applicability for modeling mixed periodic and random traffic is when $\mu_2 \gg \mu_1$ and $0 \ll c_0 \leq \rho_1 \leq 1 - c_0$, for c_0 some non-negligible portion of 1. In other words, when $C = m_1 \gg m_2 = \mu_2^{-1} (= 1, \text{typically})$ and $c_0 \leq CH^{-1} \leq 1 - c_0$ a completion time will rarely contain more than one periodic execution time. In order for this to occur, we must have some $X_{2,n} > (H - C)$ which occurs with probability $e^{-\mu_2(H-C)}$. For example, suppose $0.25 \leq \rho_1 \leq 0.75$ and $C \geq 12m_2$ ($= 12 \cdot 1$ when $\mu_2 = 1$). Then, $2.3 \cdot 10^{-16} < e^{-\mu_2(H-C)} \leq 0.0184$.[1]

When queue lengths are typically long, and a completion time contains only one blocking time, then when blocking occurs it will often be for the full blocking time B of the periodic traffic class. In the background aperiodic server discipline, $B = C$. In the foreground aperiodic server discipline, $B < C$, and sometimes it is much less. Let X be a random variable with the distribution of an aperiodic task's completion time. Under these conditions, the approximate mean and variance of X

[1]When a completion time routinely contains multiple complete periodic compute times, a degraded server model (described in Chapter 4) works reasonably well.

are given in Equation 3.2 where $\lambda = \lambda_1 + \lambda_2$.

$$\begin{array}{lll} E[X] & \approx \frac{\lambda_1}{\lambda} E[X_1] + \frac{\lambda_2}{\lambda} E[X_1 + X_2] & = m_1 + \frac{\lambda_2}{\lambda} m_2 \text{ and} \\ \mathrm{Var}(X) & = \sigma_{b,1}^2 + \frac{\lambda_2^2}{\lambda^2} \sigma_{b,2}^2 & = \frac{\lambda_2^2}{\lambda} \sigma_{b,2}^2. \end{array} \tag{3.2}$$

3.3 Poisson Arrival Processes

In all literature reviewed, when developing an exact analysis for priority queues, a number of commonly made assumptions were used for tractability. First, all input streams were assumed Poisson (each with a constant arrival rate).[2] Second, priorities are usually fixed (not dynamic) and within each class, service is typically FIFO. Third, service times are independent of time, system state, and of the arrival processes. Also, the service time density is often assumed to be absolutely continuous with respect to Lebesgue measure. Many of these assumptions can be relaxed when going to heavy traffic approximations, but are necessary when considering an exact model if any tractable solution is to ensue.

The past 30 years have been explosive in terms of developing new models for networking behavior (which can sometimes be modeled as either processor sharing or stages of servers). Priority queues have been found to be useful for efficiently scheduling periodic tasks with hard deadlines ([27], [23]). In recent years, considerably less effort appears to have been expended on finding exact probabalistic solutions for priority queues. Using either approximating models (such as heavy traffic) for which analysis is possible, or applying numeric methods for solutions to specific parameter settings appears to be much more commonplace.

Jaiswal [17] solves for the standard performance metrics when all arrival processes are Poisson and service time densities are general, except with infinite support

[2] Alternatively, all service time distributions are exponential. These memoryless assumptions lead to independence among idle and busy intervals.

and absolutely continuous with respect to Lebesgue measure. He considers different preemptive disciplines (e.g. resume, repeat) as well as non-preemptive service disciplines. In all cases, when there are two traffic classes, solutions are found in terms of LaPlace transforms (much like for the M/G/1 queue). When the service discipline is non-preemptive, FIFO within each class, and service and interarrival times are independent and exponential, the LaPlace transform for the waiting time distribution for each class is known [31].

An application of Lemma C.2.1 allows us to snapshot system performance metrics only at the arrival times of a priority class with Poisson arrivals, from which system behavior can be inferred at all other time points, even when other (higher priority) traffic does not have Poisson arrivals. Note also that for each class with Poisson arrivals, the virtual waiting time distribution is the same as the steady state waiting time distribution.

3.4 Heavy Traffic Approximations for Preemptive Fixed Priority Queues

In this section we describe heavy traffic approximations for a preemptive resume fixed priority queue. In many ways, the development parallels the heavy traffic approximation for a queue with a single traffic class so we focus primarily on defining the diffusion coefficients. These definitions are from [37]. In theory, this model describes our **background aperiodic priority** model. In practice, we will find that the model is often too optimistic (i.e. at system utilization $\rho \leq 0.99$).[3] The development here

[3] As $\rho \uparrow 1$, the HTM approximations will improve. However, in practice one would not choose $\rho = 0.9999999$ for a nominal value, for example. In Chapter 5 we propose a practical criterion which can be used to define conditions of "heavy traffic". Later in this Chapter we also consider transient overloads where $\rho > 1$ for short periods of time, but long term $\rho < 1$.

does not apply to our **foreground aperiodic priority** model, since task priority is dynamic.

For fixed priority systems, only the lowest priority class is in heavy traffic. For all priority classes other than the lowest, it turns out that queue lengths (and hence waiting times) all converge to zero when using heavy traffic theory. For the lowest priority traffic class, Whitt [37] finds that both system size and traffic load (i.e. waiting work) processes converge weakly to Brownian motion with a reflecting lower barrier at zero. The steady state distribution of the processes are found using Equation C.3.1 in Appendix A.

3.4.1 The System Size Process

High priority (periodic) tasks are class 1 and the low priority (aperiodic) tasks are class 2. The appropriate drift coefficient for the number of aperiodic tasks in the system is $m = (\lambda_2 - \mu_2(1-\rho_1))$. The notation is introduced in Table A.2. Define the n^{th} system size random function by

$$\frac{N_2^n(nt) - (\lambda_2 - \mu_2(1-\rho_1))nt}{n^{\frac{1}{2}}} = A_2^n - (1-\rho_1)B_2^n + \mu_2 X_1^n. \tag{3.3}$$

Under suitable conditions of heavy traffic, weak convergence arguments (see [37] and [4]) give

$$A_i^n \Rightarrow A_i = (\lambda_i^3 \sigma_{a,i}^2)^{\frac{1}{2}} \xi^{A_i},\ S_i^n \Rightarrow S_i = \sigma_{b,i} \xi^{S_i},\ B_i^n \Rightarrow B_i = -\mu_i^{\frac{3}{2}} S^i \sim -\mu_i^{\frac{3}{2}} \sigma_{b,i} \xi^{B_i}$$

and

$$X_i^n \Rightarrow X_i = \lambda_i^{\frac{1}{2}} S_i + \mu_i^{-1} A_i = (\lambda_i \sigma_{b,i}^2)^{\frac{1}{2}} \xi^{S_i} + (\mu_i^{-2} \lambda_i^3 \sigma_{a,i}^2)^{\frac{1}{2}} \xi^{A_i},$$

for independent Weiner processes ξ^{A_i}, ξ^{S_i}, and ξ^i, for $i \in \{1, 2\}$.

By repeated application of Lemma C.3.2, we have

$$X_i^n \Rightarrow (\lambda_i \sigma_{b,i}^2 + \mu_i^{-2}\lambda_i^3 \sigma_{a,i}^2)^{\frac{1}{2}} \xi^i$$

and the sought after result

$$\frac{N_2^n(nt) - (\lambda_2 - \mu_2(1-\rho_1))nt}{n^{\frac{1}{2}}} \Rightarrow c_A \xi^{A_2} + c_N \xi^{S_2} + c_X \xi^1 \Rightarrow (c_A^2 + c_N^2 + c_X^2)^{\frac{1}{2}} \xi \tag{3.4}$$

where the $\xi's$ are independent Weiner processes, and

$$c_A^2 = \lambda_2^3 \sigma_{a,2}^2,\ c_N^2 = (1-\rho_1)^2 \sigma_{b,2}^2 \mu_2^3,\ \text{and}\ c_X^2 = \mu_2(\lambda_1 \sigma_{b,1}^2 + \mu_1^{-1}\lambda_1^3 \sigma_{a,1}^2).$$

In our case, $X_1 \sim \mathcal{D}(C), T_1 \sim \mathcal{D}(H), X_2 \sim \mathcal{E}(\mu)$, and $T_2 \sim \mathcal{E}(\lambda)$, we have $\sigma_{b,1}^2 = \sigma_{a,1}^2 = 0, \sigma_{b,2}^2 = \mu_1^{-2}$, and $\sigma_{a,2}^2 = \lambda_1^{-2}$. Equation 3.4 then becomes

$$\frac{N_2(nt) - (\lambda_2 - \mu_2(1-\rho_1))nt}{n^{\frac{1}{2}}} \Rightarrow N_2^\infty \sim (\lambda_2 + \mu_2(1-\rho_1)^2)^{\frac{1}{2}} \xi. \tag{3.5}$$

Equation 3.6 gives the heavy traffic approximation for the steady state distribution of aperiodic system size (i.e. of $N_2^\infty = N_2$).

$$P(N_2 \leq x) = 1 - \exp(-[\frac{2(\mu_2(1-\rho_1) - \lambda_2)}{(\mu_2(1-\rho_1)^2 + \lambda_2)}]x). \tag{3.6}$$

Notice when there is no periodic traffic, $\rho_1 = 0$ and Equation 3.6 agrees with our results from the previous chapter.

3.4.2 The Virtual Waiting Time Process

With preemptive priority systems, the virtual waiting time process at priorities other than the highest is not the same as the workload process since higher priority tasks can arrive in the future which causes increased waiting times for lower priority tasks. Recall also for a preemptive fixed priority scheduling discipline, when the total system is in a state of heavy traffic, only the lowest priority class will be in heavy traffic for $\rho \leq 1$.

We begin by looking at the *workload process*, from which an approximation for the waiting time can be obtained. The notation we introduce in this section is also included Table A.2. We explicitly include here some of the constructions we use when developing state variable process simulations for response time estimation under intermediate hyperperiods. Equation 3.7 defines the random functions used.

$$\begin{aligned}
A_i(t) &= \max\{k \in \{0,1,2,...\} \mid \textstyle\sum_{j=1}^{k} T_{i,j} \leq t\} && \text{for } i \in \{1,2\} \\
X_i(t) &= \textstyle\sum_{j=1}^{A_i(t)} X_{i,j} && \text{for } i \in \{1,2\} \\
Y_1(t) &= X_1(t) - t \\
Y_2(t) &= X_2(t) + \inf_{0 \leq s \leq t} Y_1(s) \\
I_i(t) &= -\inf_{0 \leq s \leq t} Y_i(s) && \text{for } i \in \{1,2\} \\
W_i(t) &= Y_i(t) - \inf_{0 \leq s \leq t} Y_i(s) && \text{for } i \in \{1,2\} \\
X_i^n &= n^{-\frac{1}{2}}[X_i(nt) - \rho_i nt] && \text{for } i \in \{1,2\}.
\end{aligned} \tag{3.7}$$

Under suitable heavy traffic conditions, the sought result for aperiodic class 2 workload is (see [37] and [4])

$$\frac{W_2^n(nt) - (\rho_1 + \rho_2 - 1)nt}{n^{\frac{1}{2}}} \Rightarrow X_1 + X_2, \tag{3.8}$$

where just as before,

$$X_i^n \Rightarrow (\lambda_i \sigma_{b,i}^2 + \mu_i^{-2} \lambda_i^3 \sigma_{a,i}^2)^{\frac{1}{2}} \xi^i,$$

so Equation 3.8 becomes

$$\frac{\lim_{n\to\infty} W_2^n(nt) - (\rho_1 + \rho_2 - 1)nt}{n^{\frac{1}{2}}} \sim (\sigma_1^2 + \sigma_2^2)^{\frac{1}{2}} \xi, \tag{3.9}$$

when ξ is a Weiner process and $\sigma_i^2 = \lambda_i \sigma_{b,i}^2 + \mu_i^{-2} \lambda_i^3 \sigma_{a,i}^2$ for $i \in \{1, 2\}$.

When $m = \rho_1 + \rho_2 - 1 < 0$, the (class 2) workload process is positive recurrent and a steady state distribution exists. Application of Equation C.3.1 gives the limiting probability for the virtual waiting time of a class 2 job which is shown in Equation 3.10.

$$\Pr(W_2^\infty = W_2 \leq x) = 1 - e^{2m_2 x \sigma^{-2}} \tag{3.10}$$

For the specific case where $A_1 \sim \mathcal{D}(\lambda_1^{-1})$, $B_1 \sim \mathcal{D}(\mu_1^{-1})$, $A_2 \sim \mathcal{E}(\lambda_2)$, and $B_2 \sim \mathcal{E}(\mu_2)$, diffusion coefficients become $\sigma_1^2 = 0$, $\sigma^2 = \sigma_2^2 + \sigma_1^2 = 2\lambda_2 \mu_2^{-2}$, and $m = \rho - 1$. Applying these values to Equation 3.10 gives the class 2 workload distribution shown in Equation 3.11.

$$\Pr(W_2^\infty \leq x) = 1 - e^{-(1-\rho)\mu_2^2 x (\lambda_2)^{-1}} \tag{3.11}$$

Equation 3.11 describes the distribution of *pending aperiodic* work, which is *not* the virtual aperiodic waiting time since it does not include the amount of pending periodic work, and more importantly it does not take into account *future arrivals with periodic work* that will preempt pending aperiodic traffic and increase response times. Since the (time averaged) future of the periodic stream is known, the effects of current blocking times and future preemptions can be approximated.

Another application of the PASTA property (Lemma C.2.1) gives us that the pending aperiodic work distribution is that of the virtual waiting time distribution, in the absense of any future blocking by currently present or newly arriving periodic tasks. To compensate approximately for future blocking of aperiodics, pending aperiodic work is merely inflated by $(1-\rho_1)^{-1}$. This adjustment is shown in Equation 3.12. In the next chapter we will see how well and under what conditions this approximation works.

$$R(x) = 1 - e^{-(1-\rho)(1-\rho_1)\mu_2(\rho_2)^{-1}x} \tag{3.12}$$

On departure from this chapter we note that there may be long but transient durations when $m = m_w(t) \geq 0$, in which case the virtual waiting time process for the aperiodic traffic class, $W_2(t)$, can be approximated (in the long transient window) as $\mathcal{N}(m_w t, \sigma_w^2 t)$.

Chapter 4

On Performance Models

This chapter introduces the *degraded server model* (DSM) and compares it to our previously studied heavy traffic (HT) models. We will make frequent use of the DS and HT models in subsequent chapters. First, we elaborate upon our analysis objectives.

4.1 Analysis Objectives

Since we are guaranteed that periodic tasks will meet their deadlines, our focus is on the performance of aperiodic tasks. Suppose each aperiodic task has a deadline that is constant relative to its arrival time. For example, if the constant deadline is d,[1] then for each task τ_n arriving at time A_n, τ_n's (implied) deadline is at time $A_n + d$. For a (within class) FIFO server model, a desirable objective is to have $R(d) = P[R \leq d] \geq 1 - \epsilon$, for some small non-negative ϵ, where R is the aperiodic task response time variable (or cdf). An alternative objective might be to minimize $R(d)$. When d is arbitrary, these two objectives are essentially the the same. In both cases we seek to characterize the tails of the response time (or system size) distribution.

For our estimates to be applicable in practice, we seek a lower bound on response times. In other words, if R_e is our response time estimate and R_o is our observed estimate, ideally $\Pr(R_e \leq r) \leq \Pr(R_o \leq r)\ \forall\ r$. We did not attempt to

[1] In future work, the possibility of each task having a randomly chosen (non-constant) deadline is considered.

provide a strict bounding estimate for response time distributions, but we focus inspection on values of r that constitute the right tail of the distribution, since these are the critical values that define limits in a system's design.

Much emphasis is also placed on the analysis of a system that is near saturation (i.e. when $\rho = \rho_1 + \rho_2 \approx 1$) since a more lightly loaded system will perform no worse than a heavily loaded system (for a given sample path). In other words, the response times distribuion of a system with $\rho = 0.99$ will be stochastically larger than that of a system with $\rho = 0.85$ (*i.e.* $R_{\rho=0.99}(r) \leq R_{\rho=0.85}(r)\ \forall\ r$). We also consider the applicability of models when the system is not near saturation. It turns out the ratio of mean aperiodic service time to the hyperperiod has a predominant effect on the type of analysis used to describe the data when the C periodic time units are requested all at once.[2] The magnitude of $H/\mu_2^{-1} = H\mu_2$ is a significant factor in determining the appropriateness of different modeling techniques. We later characterize ranges for λ_2 as a function of $H\mu_2 (= H, \text{for } \mu_2 = 1)$ and $\rho_1 = CH^{-1}$.

4.2 Degraded Server Models (DSMs)

We did not find the phrase *degraded server model* in the referenced literature, but it is such a natural model adaptation, that its use in practice is probably not uncommon. The idea behind a degraded server model is to simply degrade the service rate by the proportion of time it is unavailable for service to a particular traffic class.

For an aperiodic service rate of μ_2, and a periodic utilization of ρ_1, the degraded aperiodic service rate can be modeled by $\tilde{\mu} = \mu_2(1 - \rho_1)$, making the mean service time appear longer. In other words, assume $B_2 \sim \mathcal{E}(\tilde{\mu})$. The degraded server response

[2] We do not permit the case of a total of C time units requested in many increments over a period of H time units. In particular, when n requests are made each for C/n time units, this is the same as defining $H' = H/n$ and $C' = C/n$.

time model is then the M/M/1 response time model with arrival rate $\lambda = \lambda_2$ and degraded service rate $\mu = \tilde{\mu}$.

When considering the aperiodic traffic in isolation, we have an M/M/1 queue, for which the response time distribution is known and given by

$$R(x) = P[R \leq x] = 1 - e^{-(\mu_2 - \lambda_2)x} = 1 - e^{-\mu_2(1-\rho_2)x}.$$

When factoring in the periodic traffic, the degraded server model approximate response time distribution becomes

$$R_d(x) = P[R \leq x] = 1 - e^{-(\tilde{\mu} - \lambda_2)x} = 1 - e^{-\mu_2(1-\rho_1-\rho_2)x} = 1 - e^{-\mu_2(1-\rho)x}. \tag{4.1}$$

Reference to the degraded server model for response times is reference to the response time cdf given in Equation 4.1. For background aperiodic service, we will find that the DSM provides a good estimator when the hyperperiod is short, but is overly optimistic when the hyperperiod is too long. For foreground aperiodic service, we will find that the DSM provides a good estimator when the hyperperiod is short, but is overly pessimistic when the hyperperiod is too long.

One thing to notice about the degraded server model is that it does not preserve utilization in the sense that $\lambda/\tilde{\mu} \neq \rho_1 + \rho_2 = \rho$. For example, when $\rho = 0.95$, $\tilde{\rho} = \lambda/\tilde{\mu} = 0.80, 0.90$ and 0.93, respectively for $\lambda = 0.2, 0.45$ and 0.70.

The same degraded server substitution also gives an approximating model for the queue length distribution which is shown in Equation 4.2.

$$N_d(x) = P[N_2 \leq n] = 1 - \tilde{\rho}^{(n+1)} = 1 - (\frac{\rho_2}{(1-\rho_1)})^{(n+1)}. \tag{4.2}$$

The degraded server system size model is a good estimator of the number of aperiodics

in the system under the same conditions that the degraded server response time model is also a good estimator for response times when using FIFO within the aperiodic task stream.

4.3 Heavy Traffic *vs* Degraded Server Models

In the previous chapter, we developed a response time estimate approximation for background aperiodic service under conditions of heavy traffic. Equation 4.3 repeats that estimate.

$$R_h(x) = 1 - e^{-(1-\rho)(1-\rho_1)\mu_2(\rho_2)^{-1}x}. \tag{4.3}$$

In the development of Equation 4.3, a degraded server assumption was made to account for currently present and future periodic tasks. We wish to compare Equation 4.3 with the degraded server response time distribution $R_d(x)$ defined by Equation 4.1.

R_h given in Equation 4.3 was derived under the conditions that $\rho \approx 1$, and otherwise cannot be expected to be reasonable. However, when applicable, heavy traffic approximations yield a nice closed form approximation for even general aperiodic arrivals and general aperiodic service disciplines, when an exact solution is often intractable.

In contrast, R_d in Equation 4.1 can be expected to apply at all traffic loadings when the periodic compute time is suitably small compared to a typical aperiodic compute time. Later simulation results show that the DSM is much less optimistic than the HTM for moderately utilized systems (e.g. $\rho = 0.85$), and in fact provides reasonable estimates for much larger values of C provided the aperiodic queue length is long enough that most response times span multiple hyperperiods.

In this section we restrict our comparison to mostly utilized (e.g. systems with $\rho \geq 0.95$). Let $\rho = \rho_1 + \rho_2 = 1 - \epsilon$ and suppose there is enough aperiodic traffic and total traffic so that $0 < \epsilon < \rho_2$ holds. When comparing R_d with R_h, observe the range $1 < (1 - \rho_1)\rho_2^{-1} = 1 + \epsilon\rho_2^{-1} < 2$, since this is the factor in the exponential exponent that distinguishes the two response time distributions. Hence, we see that $R_d(x) \leq R_h(x)$ for all $x > 0$. Since we are interested in finding estimates that error on the side of conservatism, we (somewhat arbitrarily) choose the DSM, when either the DSM or HTM are reasonable estimators.

Let $\beta_h = \mu_2(1 - \rho)(1 - \rho_1)(\rho_2)^{-1}$ and $\beta_d = \mu_2(1 - \rho)$ be the heavy traffic and degraded server response time distribution parameters, respectively. Similarly, let $x_{0.99,h}$ and $x_{0.99,d}$ be the 99^{th} percentile of the HT and DS response time distributions, respectively. Table 4.1 lists values of these variables under different system and aperiodic utilizations when $\mu_2 = 1$. The relative error, $|x_{0.99,h} - x_{0.99,d}|/x_{0.99,d}$ is also included. It increases as ρ decreases.

$\rho = 1 - \epsilon$	ρ_2	$1 + \epsilon(\rho_2)^{-1}$	β_h	β_d	$x_{0.99,h}$	$x_{0.99,d}$	error
0.99	0.25	1.040	0.0104	0.01	442.80	460.52	0.038
0.95		1.200	0.0600	0.05	76.75	92.10	0.167
0.90		1.400	0.1400	0.10	32.89	46.05	0.286
0.99	0.75	1.013	0.0101	0.01	455.96	460.52	0.010
0.95		1.067	0.0533	0.05	86.40	92.10	0.062
0.90		1.133	0.1133	0.10	40.65	46.05	0.117

Table 4.1: Coefficients for HT and DS Models

4.4 Sampling Techniques

We used two instantiations (i.e. streams generated by two different seeds) of an implementation ([12]) of George Marsaglia's universal random number generator ([28]).

Each sequence of uniform random numbers was transformed to a sequence of exponential random numbers which define the aperiodic interarrival and service times. Two sequences were used (rather than one) in an effort to simulate the assumption of independence among interarrival and service time distributions.

For each system configuration (i.e. (λ, μ, C, H) and one of FGA or BGA) only a single simulation run was made. For a fixed (λ, μ, C, H), the same seeds were used for both the FGA and BGA runs, providing a control for comparing response times between BGA and FGA service disciplines. Otherwise different seeds were used for each system configuration.

Since the aperiodic arrival process is Poisson, sampling of aperiodic response times occurs at departures. (See Lemma 2.3.2 for a justification.) There is considerable correlation in response times between adjacent arrivals. There can also be considerable correlation between adjacent hyperperiods. The response time sample size is 1000. After an initial transient period, every 1673^{rd} task's response time is observed, where 1673 seemed large to avoid correlated observations (often by skipping samples from nearby hyperperiods) for short and moderate length hyperperiods. The average number of task response times sampled per hyperperiod is $H/(1763/\lambda) = (H\lambda)/1763$. For $H = 8$ and $\lambda = 0.24$, this is roughly one response time every 900 hyperperiods. For the largest hyperperiod we considered this is less than $(0.74)32768/1763 < 14$ tasks sampled per (one long, $H = 32,768$) hyperperiod. As we will see, large hyperperiods act as probabilistic replicas, so this spacing gives in excess of 70 "nearly-independent" hyperperiods sampled. For the foreground aperiodic scheduling discipline, after some initial transience a minimum of 1500 hyperperiods were observed to obtain estimates of parameters used to define the blocking time distribution.

Before looking at our mixed periodic and aperiodic scheduling problems, we compare simulation response time EDFs of an M/M/1 simulation with their (known) theoretical response time CDFs. This comparison provides a *null* case for differences

we might (reasonably) expect to see between predicted CDFs and observed EDFs for mixed scheduling response times.

4.5 Response Time Plots

For an M/M/1 queue, the variances of both the response time and system size are proportional to $(1-\rho)^{-2}$. Consequently, when $\rho \approx 1$ a great deal of variability can be expected among sample paths, many deviating potentially significantly from the theoretical CDF. The theoretical response time CDF, $1-e^{-(\mu-\lambda)x}$, might be thought of as the "average" of a number of EDFs. Figure 4.1 shows several response time plots for a purely aperiodic (M/M/1) system when $\rho = 0.95$ (the left hand side) and $\rho = 0.99$ (the right hand side). These are examples of the *null* case, and suggest an amount of variability we might reasonably expect to see when comparing empirical response time distributions from our mixed periodic and aperiodic scheduling policies against our predicted response time distributions. Based on visual inspection, all three of the response time curves when $\rho = 0.95$ appear *reasonably* close, where as only two of the three when $\rho = 0.99$ appear to fit well. However, comparing CDFs does not emphasize differences in the right tails of the distribution. So this is just one type of comparsion we will make.

4.6 Q-Q Plots

In the previous section, our means of evaluating model fit was via visual of inspection empirical response time CDFs overlayed on the model selection(s). In this section we describe Q-Q plots and illustrate their use as a means to better compare the right tails of the distributions to the predicted models in the null cases presented above.

Let R be a theoretical response time distribution, and let E be an empirical

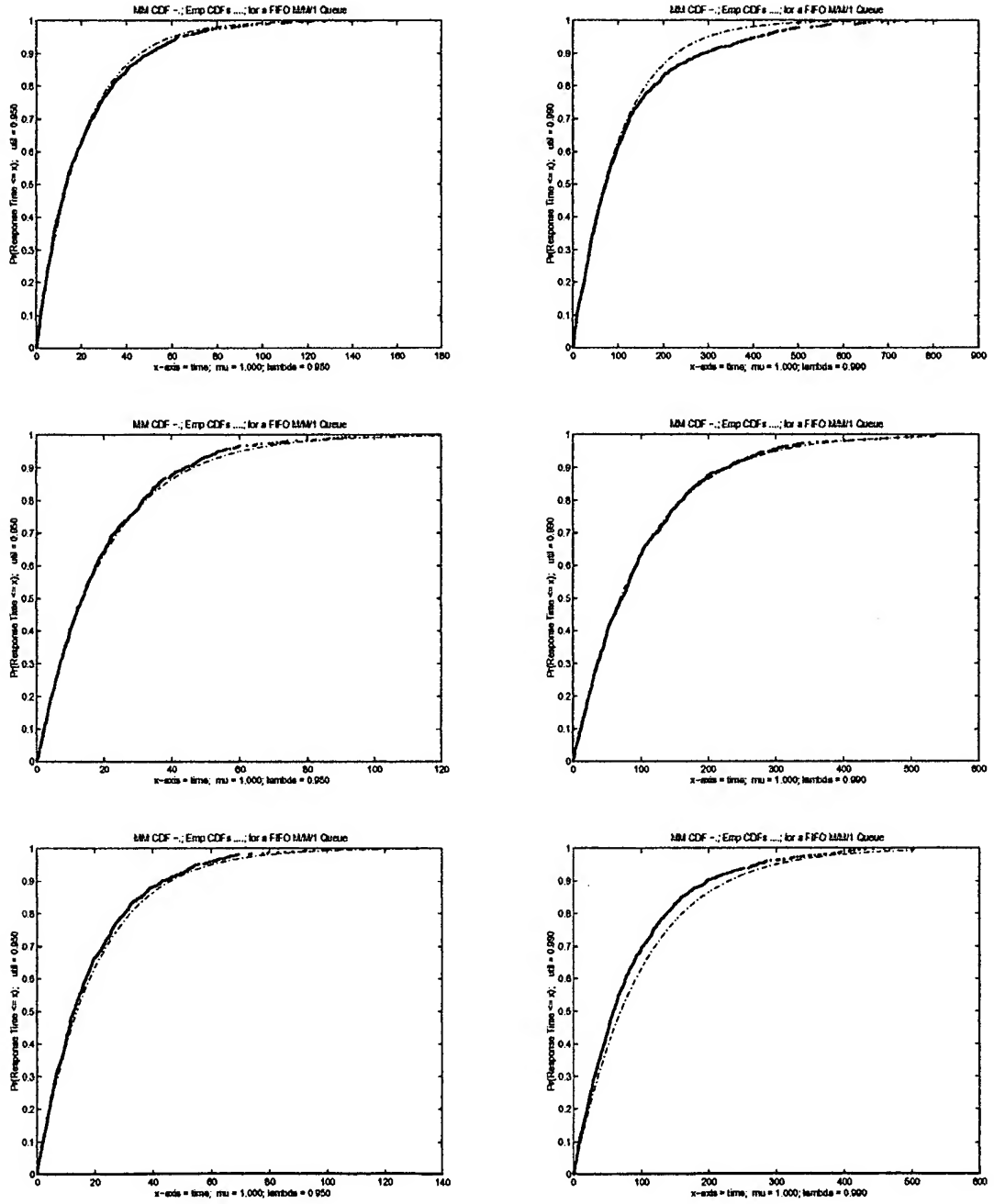


Figure 4.1: M/M/1 Response Time EDFs

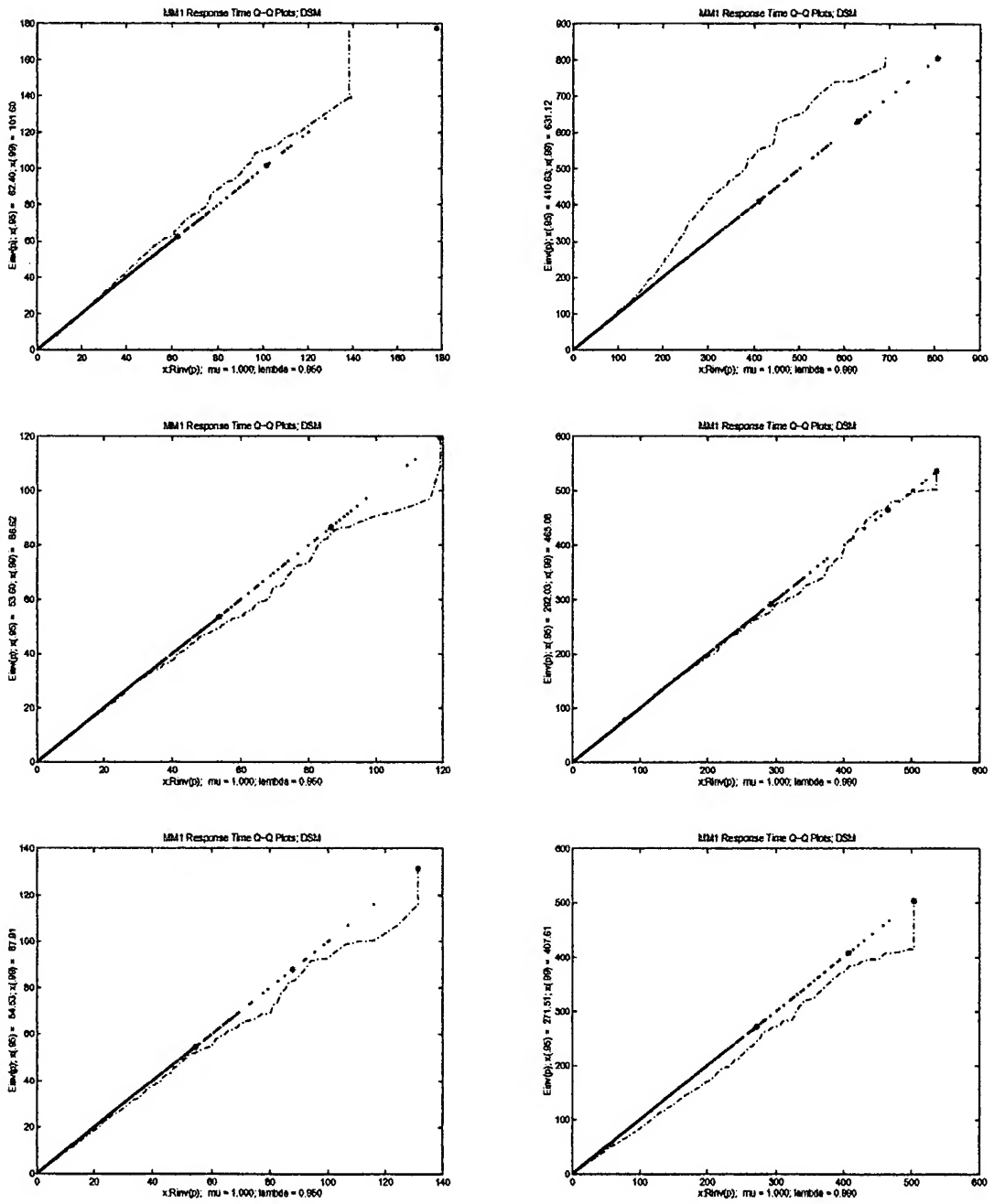


Figure 4.2: M/M/1 Response Time Q-Q Plot

response time distribution. A Q-Q plot has as its y-axis, $E^{-1}(p)$ for $0 \leq p \leq 1$ and $R^{-1}(p)$ for the x-axis. Ideally, a Q-Q plot will produce the line $y = x$. Observed differences in the right tails will be larger than observed differences in the left tails, since the response time CDF domain values increase as $p \uparrow 1$. Since we want the empirical CDF to be stochastically larger than the predicted CDF, in the case of a mismatch, the preference is for the Q-Q plot output to fall below (i.e. to the right of) the line $y = x$.

The Q-Q plots corresponding to the response time curves in Figure 4.1 are shown in Figure 4.2. As expected, the largest differences tend to occur in the right tails. The inverse of the theoretical M/M/1 response time curve involves taking the log of a number near zero, giving rise to an infinite slope at $p = 1.0$.

The asterisks are placed at values for which $p = 0.95$, $p = 0.99$ and $p = 1.0$. For $\rho = 0.95$ (the left hand side), the agreement is fairly good to the 95^{th} percentile. Between the 95^{th} and 99^{th} percentile, the agreement is still moderately good, and typically there is one or more noticable points of departure between the 99^{th} and 100^{th} percentile. In all three cases the maximum relative error appears to be between 0.20 and 0.25. For $\rho = 0.99$ (the right hand side), one of the samples is in much closer agreement than all three of the samples when $\rho = 0.95$. In the remaining two samples, divergence away from $y = x$ begins well before the 95^{th} percentile. In these two cases, the maximum relative error again appears to be between 0.20 and 0.25. These sample deviations for the null case provide a basis for what deviations we might see when approximating response time distributions for our mixed periodic and aperiodic scheduling problems.

Chapter 5

Mixed Scheduling: Background Aperiodics

In this chapter, we study the behavior of the **background aperiodic service** discipline. For this discipline, priorities are fixed, with priority 1 (highest) belonging to periodic tasks and priority 2 (lowest) belonging to aperiodic tasks. We will attempt to characterize both the steady state response time and system size distributions. Most of our investigations will focus on heavily loaded systems, since it is peak traffic periods (which might also correspond to long periods of transient overload) that are of concern in real-time systems. We are also primarily interested in (conservative) estimation of the right tails of the distributions.

5.1 System Specification

Given the scheduling discipline, there are four parameters that define the system specification:

1. the periodic task stream's interarrival rate or equivalently, the time between periodic arrrivals ($H = \lambda_1^{-1}$),

2. the periodic compute time ($C = \mu_1^{-1}$), [1]

[1] H stands for Hyperperiod. When only a single periodic task is present, H is its period. When there are multiple periodic streams present (discussed in Chapter 7), H is defined to be the least common multiple of the periods of all periodic tasks. Similarly, C stands for task Compute time. This notation has become standard in real-time scheduling theory for periodic tasks and we will often make use of these mnemonics.

3. the aperiodic task stream's interarrival rate $\lambda = \lambda_2$ and

4. the aperiodic service rate $\mu = \mu_2$. Unless otherwise stated, $\mu_2 = 1$ in the reported simulation results. Problems with different values of μ_2 can be scaled.

In turn, these parameters define:

1. the periodic utilization ($\rho_1 = \lambda_1/\mu_1 = C/H$),

2. the aperiodic utilization ($\rho_2 = \lambda_2/\mu_2 = \lambda/\mu$) and

3. the system utilization ($\rho = \rho_1 + \rho_2$). Under conditions of heavy traffic, the system is near saturation (i.e. $\rho \approx 1$). We mostly consider values of $\rho = 0.99$ and $\rho = 0.95$ since heavy traffic conditions provide bounding curves for lighter traffic conditions.

5.2 Aperiodic Response Time Analysis

We will find that different models provide better estimators for different parameter specifications. In particular, for fixed μ_2, ρ_2, and ρ, the models change as H (and hence C) increase. When comparing the foreground and background aperiodic service disciplines, it is helpful to introduce the concept of a **periodic blocking time**. The periodic blocking time is defined as the time a periodic task executes while one or more aperiodic tasks waits. We let $\bar{B}$ denote the average blocking time.

Under conditions of heavy traffic, for the foreground aperiodic discipline, $\bar{B} \approx C$. When $\mu_2^{-1} \gg H$, a single aperiodic task will likely experience more than a single blocking time during its completion time. In this case, the degraded server model is (intuitively) reasonable, and in fact provides fairly good estimates. Under these conditions, the parameters define a *short hyperperiod model* (SHM).

When $\mu_2^{-1} \ll C < H$, then only a small percentage of aperiodic tasks will experience a blocking time as a part of their completion time. For C sufficiently large, using only first moments, a good *fluid flow* estimator can be derived. As λ_2 (and ρ) decrease, this estimator remains valid, and C can also be decreased. Under these conditions, the parameters define a *long hyperperiod model.*

The remaining case, the *intermediate hyperperiod model,* poses the greatest analytic challenges. In this case, we will see that response times fall in distinct *bands* which can be used to define response time ranges. For the intermediate hyperperiod model, we construct a piecewise linear CDF with response time values determined by band ranges, and CDF values determined by the observed probability of being in each band.

5.2.1 Short Hyperperiods

When $\mu_2^{-1} \gg H$, a typical aperiodic service time will span multiple hyperperiods during each of which a blocking time equal to C will occur, except possibly for the first hyperperiod execution (when an aperiodic arrives to an empty aperiodic queue during a $[0, C]$ time interval). Under these conditions, the degraded server model (DSM, see Section 4.2) was observed to work well under a range of traffic loading.

When μ_2^{-1} is not an order of magnitude or more larger than H, but when the system is sufficiently saturated (*e.g.* $\rho \geq 0.99$), aperiodic queue lengths will tend to be long, so a response time for a newly arriving task will also tend to span several hyperperiods, even though its execution time is blocked by at most one (and probably zero) periodic task execution(s). This observation is formalized in Conjecture 5.2.1. Since the DSM and HTMs are close to one another, the criterion in Conjecture 5.2.1 might be viewed as providing a practical rule of thumb for determining conditions of heavy traffic.

Conjecture 5.2.1 (BGA SHM Conditions) *For background aperiodic scheduling, when the expected number of aperiodic arrivals during an average (DSM) response time is greater than or equal to the expected number of aperiodics discharged in a hyperperiod, the DSM provides reasonable estimates.*

More concisely, the DSM is reasonable when

$$\lambda_2 \bar{R}_{\text{dsm}} = \frac{\lambda_2}{(\mu_2(1 - CH^{-1}) - \lambda_2)} \geq (H - C)\mu_2$$

or equivalently, when

$$\lambda_2 \geq \frac{\mu_2^2(1-\rho_1)^2}{H^{-1} + \mu_2(1-\rho_1)} = \frac{\mu_2(1-\rho_1)^2}{(H\mu_2)^{-1} + (1-\rho_1)}.$$

Given the applicability of the DSM, the approximate theoretical mean and variance of the response times are those of the approximating models, given in observation 5.2.2.

Observation 5.2.2 (BGA SHM response time mean and variance) *When the DSM provides reasonable estimates for the background aperiodic scheduling model, the approximate theoretical response time mean is*

$$m_R = \bar{R} = (\mu_2(1-\rho_1) - \lambda_2)^{-1},$$

and the approximate theoretical response time variance is

$$\sigma_R^2 = (\mu_2(1-\rho_1) - \lambda_2)^{-2}.$$

This follows from the exponential DSM response time distribution, $\mathcal{E}(\mu_2(1-\rho_1) - \lambda_2)$. Observe that for fixed ρ_1, the response time mean and variance do not change with

H. Similarly, the DSM response time mean and variance do not change when both ρ and μ_2 are fixed.

Table 5.1 shows sets of (H, C) values and minimum λ values for which the DSM is a decent predictor. For fixed CH^{-1}, as H increases, so must λ to ensure the queue lengths remain long much of the time. For an equilibrium distribution to exist, $\rho < 1$, so $\lambda_2 = \mu_2\rho_2$ can not be increased without bound, and the DSM is reasonable only for H not too large. Table 5.1 contains some observed values for the mean and standard deviation of sample response times.

H	C	$\lambda_{2,\min}$	$\hat{m}_r$	$\hat{\sigma}_r$
16	4	≥ 0.693	$m_r = 100$	$\sigma_r = 100$
64	16	≥ 0.735	theory	theory
128	32	≥ 0.742		
16	12	≥ 0.207	for $\rho = 0.99$	for $\rho = 0.99$
64	48	≥ 0.236		
128	96	≥ 0.242		
8	2	0.74	108.30	101.49
32	8	0.74	91.20	84.09
128	32	0.74	93.40	80.70
8	6	0.24	97.07	100.77
32	24	0.24	103.69	93.93
128	96	0.24	127.25	97.51

Table 5.1: BGA SHM Response Time Sample Moments

Figure 5.1 shows several empirical response time CDFs with a theoretical DSM overlay under a range of conditions which meet and then exceed the criteria in Conjecture 5.2.1. Figure 5.2 shows the corresponding Q-Q plots. When H becomes long, the DSM becomes much too optimistic.

In practice, this case is not one of the greatest concern since it occurs infrequently in hard real-time systems and the time an aperiodic is blocked by a periodic

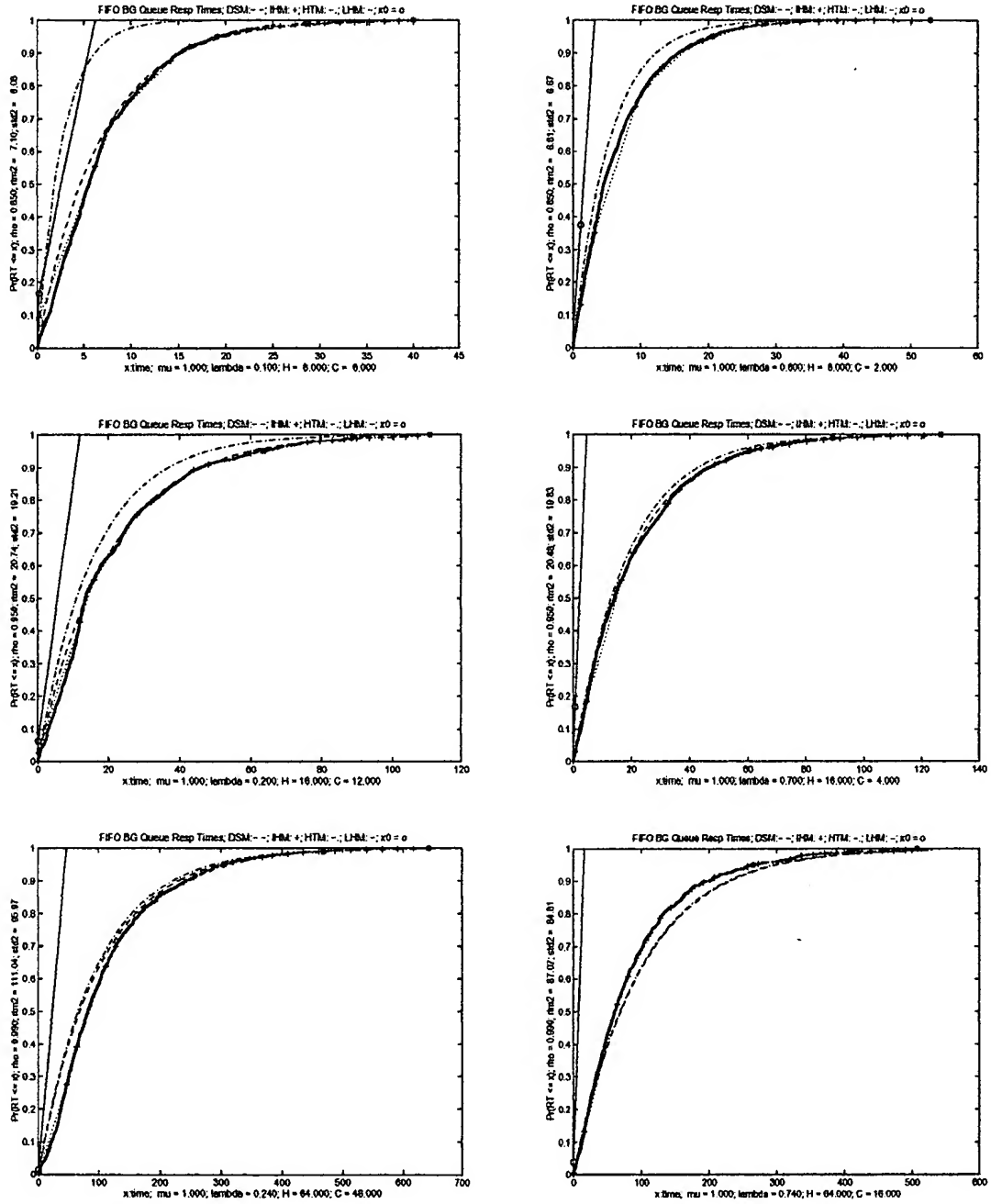


Figure 5.1: BGA SHM Response Time EDFs

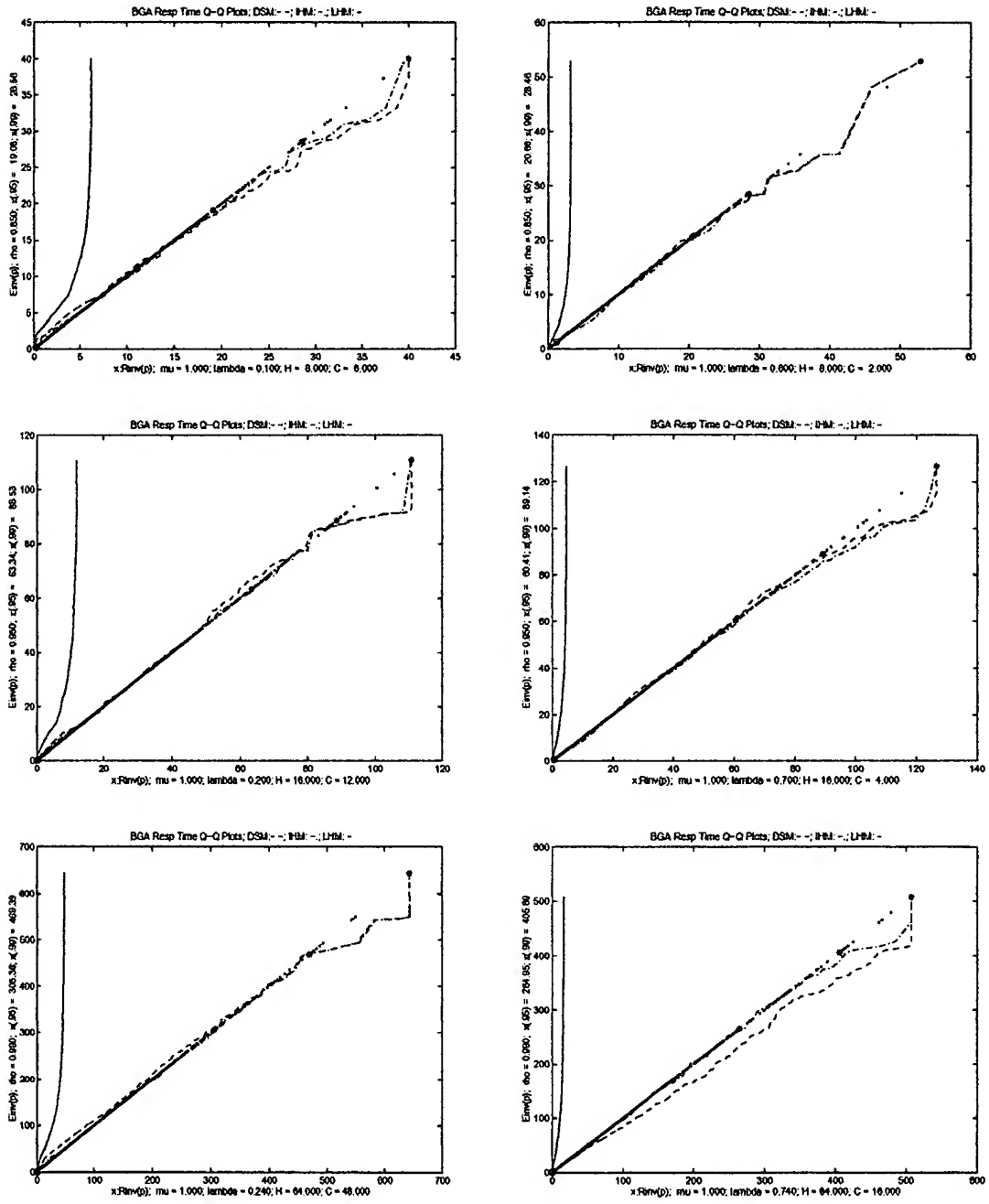


Figure 5.2: BGA SHM Response Time Q-Q Plots

can be less variable than the fluctuations in the aperiodic system size process.

5.2.2 Long Hyperperiods

When the hyperperiod is much longer than the mean aperiodic service time, we devise a *long hyperperiod model* (LHM) using only first moments to characterize the response time distribution of an aperiodic task. This model is applicable under both light and heavy traffic conditions, with an increase in H required for applicability as ρ increases. We also provide criteria for when the long hyperperiod model works reasonably well, and calculate the approximate mean and variance of aperiodic response times.

In our long hyperperiod approximation, each hyperperiod is modeled as having a queue build-up (B) period, a discharge (D) period and then an M/M/1 (M) period. In order for this approximation to be good, the hyperperiod must be long enough for the discharge to occur so at the end of the hyperperiod, the queue backlog is no longer present and the system is essentially in an M/M/1 state. The buildup and discharge periods will be smaller for lighter aperiodic traffic, and this approximation is better at shorter hyperperiods when ρ_2 is small. This approximation is not good when the build-up and discharge periods extend across multiple hyperperiods (such as in the previous section). As we will see when $\rho \approx 1$, H needs to be rather long for this approximation to be reasonable.

We now calculate our approximation for suitably long hyperperiods. Let $t \in [0, H]$ and suppose an arrival occurs at time t. Suppose further that at time t, the number of arrivals in that hyperperiod equals the expected number of arrivals in $[0, t]$, or equivalently λt, and that the number of departures in $[0, t]$ is equal to the expected number of departures in $[0, t]$ which varies as a function of t due to blocking by the periodic task.

To calculate the expected departure time of an arrival at time t, partition the

hyperperiod in three intervals:

1. At the start of the hyperperiod there are n_0 tasks.[2] In the blocking (B) interval, $[0, B]$ all new aperiodic arrivals queue, since they have lower priority than the blocking periodic task. In the case of background aperiodic tasks, the left hand limit of B is always C. We use the notation $B = C$ to more generally denote a (possibly random) blocking time (such as for the foreground aperiodic service discipline).

2. Next is the discharge (D) interval defined on $[B, \mathcal{F}]$ where $\mathcal{F}$ is the first time, since the start of the hyperperiod, that the queue contains n_0 tasks (where n_0 was the number of tasks at the start of the hyperperiod). Note also that $\mathcal{F}$ is a stopping time and $\mathrm{E}(\mathcal{F}) = (\mu B)(\mu - \lambda)^{-1}$.

3. Last is the M/M/1 (M) interval where the process has returned to an M/M/1 queue beginning in state equal to n_0. All residual effects of the blocking period introduced by the period process are gone. M $= [\mathcal{F}, H]$ and can be null when the length of B exceeds $H - B$. When M is routinely null, this approximation is poor. For suitably large H, approximating $\mathcal{F}$ by its expected value works well.

Table 5.2 introduces the notation we use to develop the long hyperperiod model's response time analysis. When the long hyperperiod model is applicable, the stochastic behavior of each hyperperiod is similar, so it suffices to study the response time behavior in a single arbitrary hyperperiod. The notation in Table 5.2 is assumed to be with respect to some arbitrary hyperperiod, and not the beginning of time. The selection of values for n_0 and x_0 are described in Section D.1.1 of the appendix.

To approximate the virtual response time of an (hypothesized) arrival at time t, the three regions are considered separately. Assume there are n_0 aperiodic tasks

[2] A technique for choosing n_0 is described in Section D.1.1.

Notation	Description
n_0	The number of aperiodic tasks in the system at time 0, the start of the hyperperiod.
x_0	The expected value of an aperiodic response time when there are n_0 aperiodic tasks in the system and no (present or new) periodic tasks. $x_0 = n_0\mu_2^{-1}$.
$\mathcal{F}$	The first passage time to the state, k at time 0 (typically $k = n_0$) since departing state k after the start of the hyperperiod. $\mathcal{F} \approx E[\mathcal{F}] = B + (\lambda B)/(\mu - \lambda) = (\mu B)/(\mu - \lambda) = B(1 - \rho_2)^{-1}$.
B	The compute time of the periodic task, τ. $B = C$, to remind us that C is a blocking time and to allow for the possibility that B is a random variable.
H	The hyperperiod, or equivalently λ_1^{-1}.
$R_B(t)$	The *virtual* response time, which is the response time assuming an arrival occurred at time $t \in \mathrm{B} = [0, B]$. For long H, $R_B(t) \approx E[R_B(t)]$.
$N_B(t)$	The expected number in the system at time $t \in [0, B]$. $N_B(t) = n_0 + \lambda_2 t$.
$R_D(s)$	The virtual response time for an arrival at time $s \in \mathrm{D} = [B, \mathcal{F}]$. The response time formulas differ based on arrival time. $R_D(s) \approx E[R_D(s)]$.
$N_D(t)$	The expected number in the system at time $t \in [B, \mathcal{F}] = D$.
$R_M(u)$	The virtual response time for an arrival at time $u \in \mathrm{M} = [\mathcal{F}, H]$.

Table 5.2: Response Time Notation in Background Mode

present at the start of the hyperperiod. The exact value of n_0 turns out not to be significant under conditions of heavy traffic.

In the blocking region, the virtual response time is approximated by its mean shown in Equation 5.1. At time t, $N_B(t) = n_0 + \lambda_2 t$. No tasks begin discharging until time B, which is $B - t$ units away. Tasks then depart at an average rate of μ_2.

$$R_B(t) \approx E(R_B(t)) = (B - t) + N_B(t)\mu_2^{-1} = (B - t) + \frac{(n_0 + \lambda_2 t)}{\mu_2} \tag{5.1}$$

For $s \in [B, \mathcal{F}]$ the process is discharging its queue backlog. At time s, the expected number of arrivals is $n_0 + \lambda_2 s$. The expected number of departures is $(s - B)\mu_2$. So the expected number in the system at time s is $E[N_D(s)] = (n_0 + \lambda_2 s) - (s - B)\mu = (n_0 + \lambda_2 B) - (s - B)(\mu_2 - \lambda_2)$. In the region $[B, \mathcal{F}]$, by hypothesis the queue never empties, so the expected virtual response time is $N_D(s)\mu_2^{-1}$, shown

in Equation 5.2. Note that $R_D(s) = R_B(s)$.

$$R_D(s) \approx E(R_D(t)) = B + \frac{(n_0 - (\mu_2 - \lambda_2)s)}{\mu_2} = (B - s) + \frac{(n_0 + \lambda_2 s)}{\mu_2} \tag{5.2}$$

Lastly, for $u \in [\mathcal{F}, H]$ the queue behaves as an M/M/1 queue, and the expected virtual response time is simply the mean of the theoretical M/M/1 response time distribution conditional on n_0 tasks in the queue which is $\mathcal{G}(\mu_2, n_0)$, which has mean

$$R_M(u) \approx E(R_M(u)) = \frac{n_0}{\mu_2} = x_0. \tag{5.3}$$

Since there is always a virtual task, the queue is never empty in the virtual response time analysis so the gamma distribution seems an appropriate approximation for response times. However, when system traffic is light, an observed queue might often empty in the M/M/1 interval. For an M/M/1 queue, the (unconditional) steady state departure process has been shown to have distribution $\mathcal{E}(\lambda_2)$ [35], which may be a better model for light system loadings. When $\mathcal{F}$ and n_0 are known, the virtual expected response time as a function of time can be graphed from Equations 5.1, 5.2 and 5.3 and is shown in Figure 5.3.

When the LHM works well, subsequent simulation plots will show that all response times will fall in the band defined by lines $[(0, C), (C, 0)]$ and $[(0, H), (H, 0)]$.

We now approximate $\mathcal{F}$. At time B, the expected number of aperiodic tasks queued is $n_0 + \lambda_2 B$. An application of Equation 2.7 leads to Equation 5.4 as our approximation;

$$\mathcal{F} \approx E[\mathcal{F}] = B + \mathrm{E}[n_0 + \lambda_2 B \rightarrow n_0] = B + (\lambda_2 B) \cdot E[\mathcal{B}] = B + \frac{\lambda_2 B}{\mu_2 - \lambda_2} = \frac{B}{(1 - \rho_2)}. \tag{5.4}$$

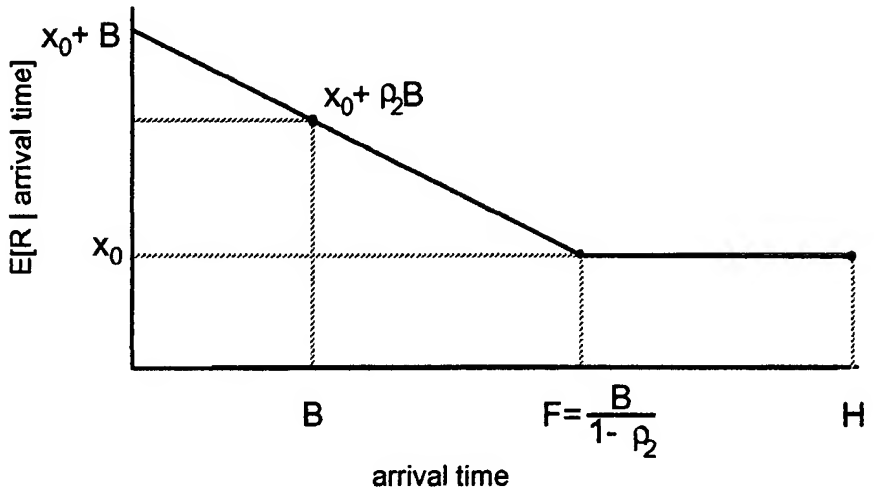


Figure 5.3: BGA LHM: E[Response Time | Arrival Time] *vs* Arrival Time

Note that $E(\mathcal{F}) \leq H$, and thus does not take into account the possibility of statistical fluctuations in the queue length process. Also the expected response time $E(R)$ is maximum when the arrival is at time zero, and $E_B(0) = B + n_0(\mu_2 - \lambda_2)^{-1}$. Both these observations emphasize that this model can only be a reasonable approximation when all arriving tasks depart within one hyperperiod and also suggests its use for light traffic.

We formalize conditions when the long hyperperiod model is a reasonable approximation in Conjecture 5.2.3.

Conjecture 5.2.3 (BGA LHM Conditions) *For background aperiodic scheduling, when the M/M/1 subinterval of the hyperperiod exceeds some multiple of the response time standard deviation in the degraded server model, the long hyperperiod (LHM) model provides reasonable estimates.*

More concisely, the LHM is reasonable when

$$x_0 + k_{\rho_1}\sigma_{\mathrm{r,dsm}} \leq H - \mathcal{F} \tag{5.5}$$

for some constant $k_{\rho_1} = k/\rho_1 > 0$, *or equivalently, when*

$$x_0 + \frac{k_{\rho_1}}{(\mu_2(1-\rho_1) - \lambda_2)} < H - \frac{C\mu_2}{(\mu_2 - \lambda_2)}. \tag{5.6}$$

If we view x_0 *as a constant (when in fact it depends on* λ_2*), and let*

$$\begin{aligned} A = &\ 1.0 - x_0 H^{-1}, \\ B = &\ 2\mu_2(\rho_1 - 1) + k_{\rho_1} H^{-1} + (\mu_2 x_0 H^{-1})(2 - \rho_1), \text{ and} \\ C = &\ (\mu_2(1-\rho_1))^2 - (k_{\rho_1}\mu_2 H^{-1}) - (x_0 \mu_2^2 H^{-1})(1-\rho_1), \end{aligned} \tag{5.7}$$

then the criterion reduces to

$$\lambda_2 \leq \frac{-(B + \sqrt{B^2 - 4AC})}{2A} \approx \mu_2(1-\rho_1) - \frac{k_{\rho_1}}{2H} - \frac{1}{2}\sqrt{\frac{k_{\rho_1}^2}{H^2} + \frac{4k_{\rho_1}\mu_2\rho_1}{H}}. \tag{5.8}$$

Equation 5.8 defines a criterion for when the LHM can be expected to give reasonable estimates for applications. To understand the intuition behind Equation 5.5, refer to Figure 5.3. Roughly speaking, the condition of Equation 5.5 can be viewed as an approximate confidence interval. Let I_m be a random variable defining the length of the M/M/1 interval (technically, $I_m = H - \mathcal{F}$). When in the discharge interval, we postulate that the queue length (and response time) behavior is either approximated by or no worse than what we would observe under the DSM conditions. At the start of the M/M/1 interval, we pessimistically assume the response time variance for the first task to begin/resume service is $\sigma^2_{r,\text{dsm}}$. This criterion is simply that the left hand response time interval limit does not exceed the M/M/1 interval duration, $H - \mathcal{F}$.

When the condition in Equation 5.8 is not met, there are likely to be intervals (e.g. transient periods) where many arrivals will not depart within H time units of

their arrival time in which case the long hyperperiod model cannot be expected to hold. Observations suggest a value of $k = 6$ works well, with $k_{\rho_1} = k(\rho_1)^{-1}$.

Table 5.3 lists some sample ranges of λ_2 for which the LHM is a reasonable estimate given the other system parameters. $k = 6$ has been chosen large enough that the λ ranges appear slightly conservative thoughout all values of H. For $\lambda_2 < \lambda_{2,\max}$ the fit is minimally good to the 99^{th} quantile or better. When $\rho_{\max} \approx 1$, the fit is good or better even when $\lambda_2 > \lambda_{2,\max}$. For example, consider $H = 16384$, and $\lambda_2 = 0.24$, where the fit is exact. The quantile column lists the quantile at which the predicted and observed diverge noticably. The fit column list a qualitative assessment of the fit upto the quantile at which divergence occurs.

H	C	$\rho_{\max}$	λ_2 max	approx. fit	quantile	λ_2
128	32	0.67	< 0.42	poor		0.60
256	64	0.79	< 0.54	moderate	x(.95)	0.60
512	128	0.87	< 0.62	good	x(.99)	0.60
512	128	0.87	< 0.62	moderate	x(.95)	0.70
1024	256	0.91	< 0.66	good	x(.97)	0.70
2048	512	0.94	< 0.69	very good	x(.99)	0.70
4096	1024	0.96	< 0.71	moderate	x(.95)	0.74
8192	2048	0.97	< 0.72	good	x(.97)	0.74
16384	4096	0.98	< 0.73	very good	x(.997)	0.74
128	96	0.75	< 0.00	good (in tails)	x(.99)	0.10
256	192	0.83	< 0.08	very good	x(.99)	0.10
512	384	0.88	< 0.13	excellent	x(1.0)	0.10
512	384	0.88	< 0.13	very good	x(.99)	0.20
1024	786	0.92	< 0.17	excellent	x(1.0)	0.20
2048	1536	0.94	< 0.19	excellent	x(1.0)	0.20
4096	3072	0.96	< 0.21	good	x(.99)	0.24
8192	6144	0.97	< 0.22	excellent	x(1.0)	0.24
16384	12288	0.98	< 0.23	exact	x(1.0)	0.24

Table 5.3: BGA LHM Response Time Criterion Evaluation

A CDF is easily constructed from Figure 5.3 and is shown in Figure 5.4. In Figure 5.3, the slope between $[0, B + x_0]$ and $[B, x_0 + \rho_2 B]$ is $(\rho_2 - 1)$. This turns out to be the same slope as between points $[B, x_0 + \rho_2 B]$ and $[\mathcal{F}, x_0]$. In the simplest construction of the long hyperperiod response time CDF, there is a point mass at x_0 with measure $1 - \mathcal{F}H^{-1}$. The jump height can be viewed as the probability an aperiodic arrival occurs in the M/M/1 interval. It can also be thought of as the probability that a randomly chosen aperiodic task experiences no blocking delays, either directly or from residual backlog. Then there is a (straight) line from $[\mathcal{F}, 1 - \mathcal{F}H^{-1}]$ to $[B + x_0, 1.0]$. The slope between x-coordinates x_0 and $B + x_0$ is $[(1 - \rho_2)H]^{-1}$, which rises slowly when H is large. An expression for the CDF shown

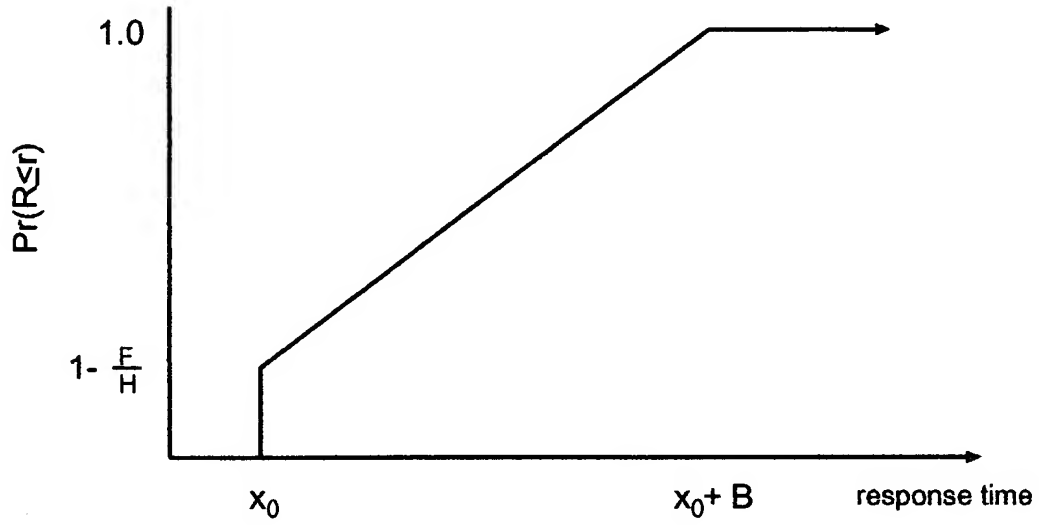


Figure 5.4: BGA LHM Predicted Response Time Distribution

in Figure 5.4 is given in Equation 5.9. Note that the quantity $1 - (x - x_0)B^{-1}$ can be interpreted as Pr(blocking time a newly arriving task experiences $\geq x - x_0$) for

$x \in [x_0, B + x_0]$.

$$R_l(x) = \begin{cases} 0 & \text{for } x < x_0 \\ 1 - \frac{\rho_1}{(1-\rho_2)} & \text{for } x = x_0 \\ 1 - \frac{\rho_1}{(1-\rho_2)}(1 - \frac{(x-x_0)}{B}) & \text{for } x_0 \leq x \leq B + x_0 \\ 1 & \text{for } x \geq B + x_0 \end{cases} \tag{5.9}$$

Using the CDF shown in Figure 5.4, the distributional moments are easily computed. Conjecture 5.2.4 lists the response time mean and variance.

Conjecture 5.2.4 (BGA LHM response time mean and variance) *When the LHM is reasonable, the mean response time is given by*

$$E[R] = x_0 + \frac{\rho_1 B}{2(1-\rho_2)} \approx \frac{\rho_1 B}{2(1-\rho_2)} \text{ for } B \gg x_0, \tag{5.10}$$

and the variance is given by

$$\text{Var}[R] = \frac{\rho_1 B^2}{\mu_2(1-\rho_2)}[\frac{1}{3} - \frac{\rho_1}{4(1-\rho_2)}]. \tag{5.11}$$

Both these calculations assume a point mass of $1 - \mathcal{F}H^{-1} = 1 - \rho_1(1-\rho_2)^{-1}$ *at point* $x_0 = n_0\mu_2^{-1}$.

Let $G(t) = P(\text{an arrival occurs in } [0,t] \mid \text{an arrival occurs in } [0,H])$. Then, by the assumption of Poisson arrivals, $G(t) = tH^{-1}$ and $g(t) = dG(t)/dt = H^{-1}$. The mean can also be computed as follows

$$\begin{aligned} E[R] &= \int_0^H E[R_t|\text{arrival at } t]g(t)dt \\ &= H^{-1}[\int_0^{\mathcal{F}}((B-s) + [n_0 + \lambda_2 s]\mu_2^{-1})ds + \int_{\mathcal{F}}^H x_0 ds] \\ &= x_0 + \tfrac{1}{2}(\rho_1 B)(1-\rho_2)^{-1}. \end{aligned} \tag{5.12}$$

Table 5.4 shows serveral examples of theoretical and observed means and standard deviations. The observed values are based on a single simulation run at the respective parameter setting.

H	C	λ_2	$\bar{r}$	$\hat{r}$	σ_r	$\hat{\sigma}_r$	$r_{\max}$
2048	512	0.74	318	246	198	153	1020
4096	1024		586	492	325	306	1375
8192	2048		1024	984	625	612	2400
2048	1536	0.24	842	758	466	449	1600
4096	3072		1540	1516	898	898	3100
8192	6144		3137	3032	1824	1797	6200

Table 5.4: BGA LHM Sample Moments

The LHM is applicable when system traffic is not "heavy". Rather than assign a point mass at $x_0 = n_0\mu_2^{-1}$, it makes sense to define x_0 to be such that response times less than x_0 are defined by the M/M/1 response time curve, and for those greater than $n_0\mu_2^{-1}$ the response time curve is defined by the long hyperperiod model curve. This not only eliminates any point masses in the response time curve, but it also increases x_0 with decreasing ρ_2 (and fixed ρ_1) corresponding to an increase in the M/M/1 interval (in contrast to decreasing $\bar{N}_2$ with decreasing ρ_2), as one would expect. This adaptation is described in Section D.1.1.

The predictions given by the CDF in Figure 5.4 are optimistic near the boundaries of LHM conditions in Equation 5.8 in which case a model for intermediate hyperperiods might provide better response time estimates. The corresponsding Q-Q plots are shown in Figure 5.6.

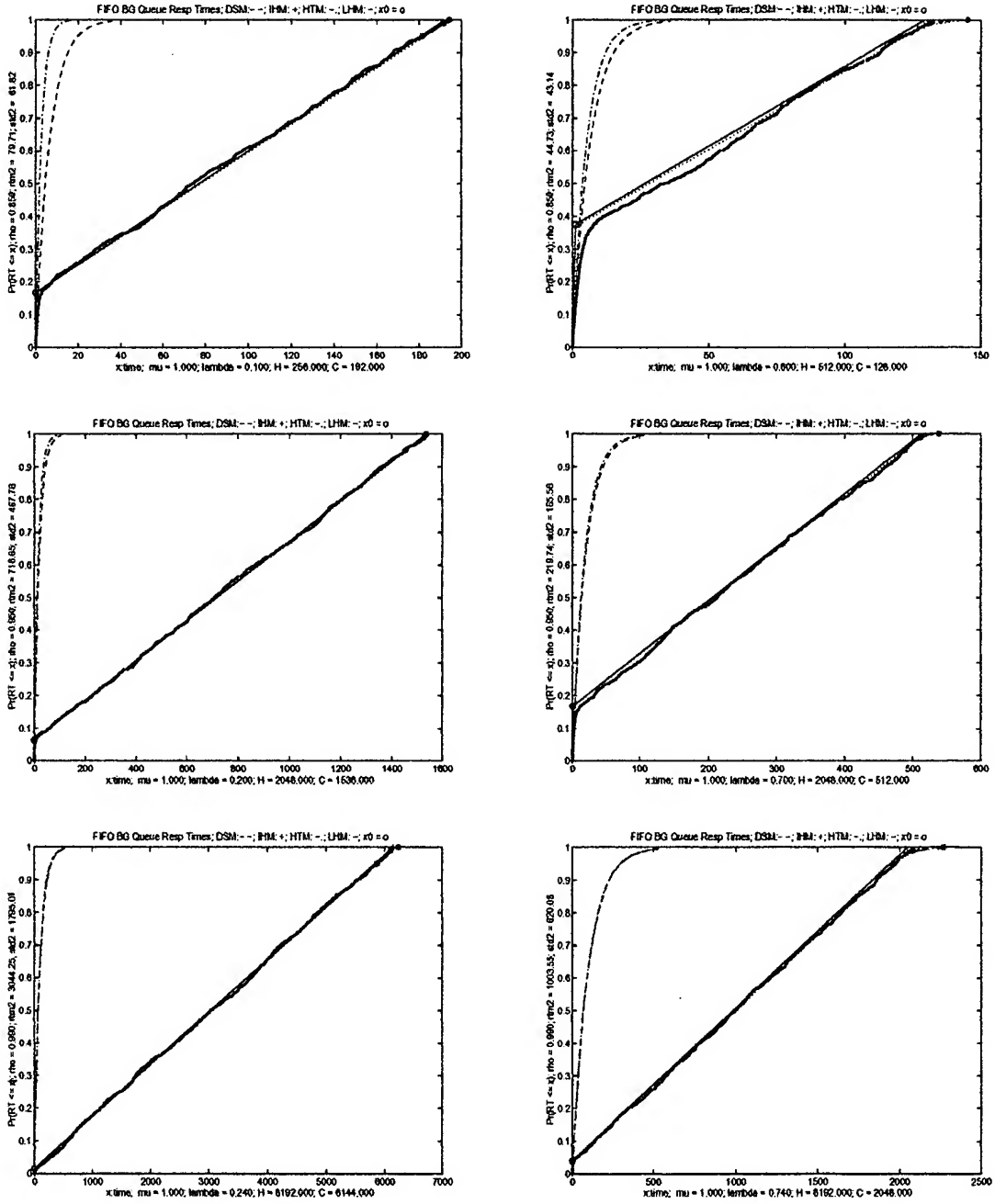


Figure 5.5: BGA LHM Response Time EDFs

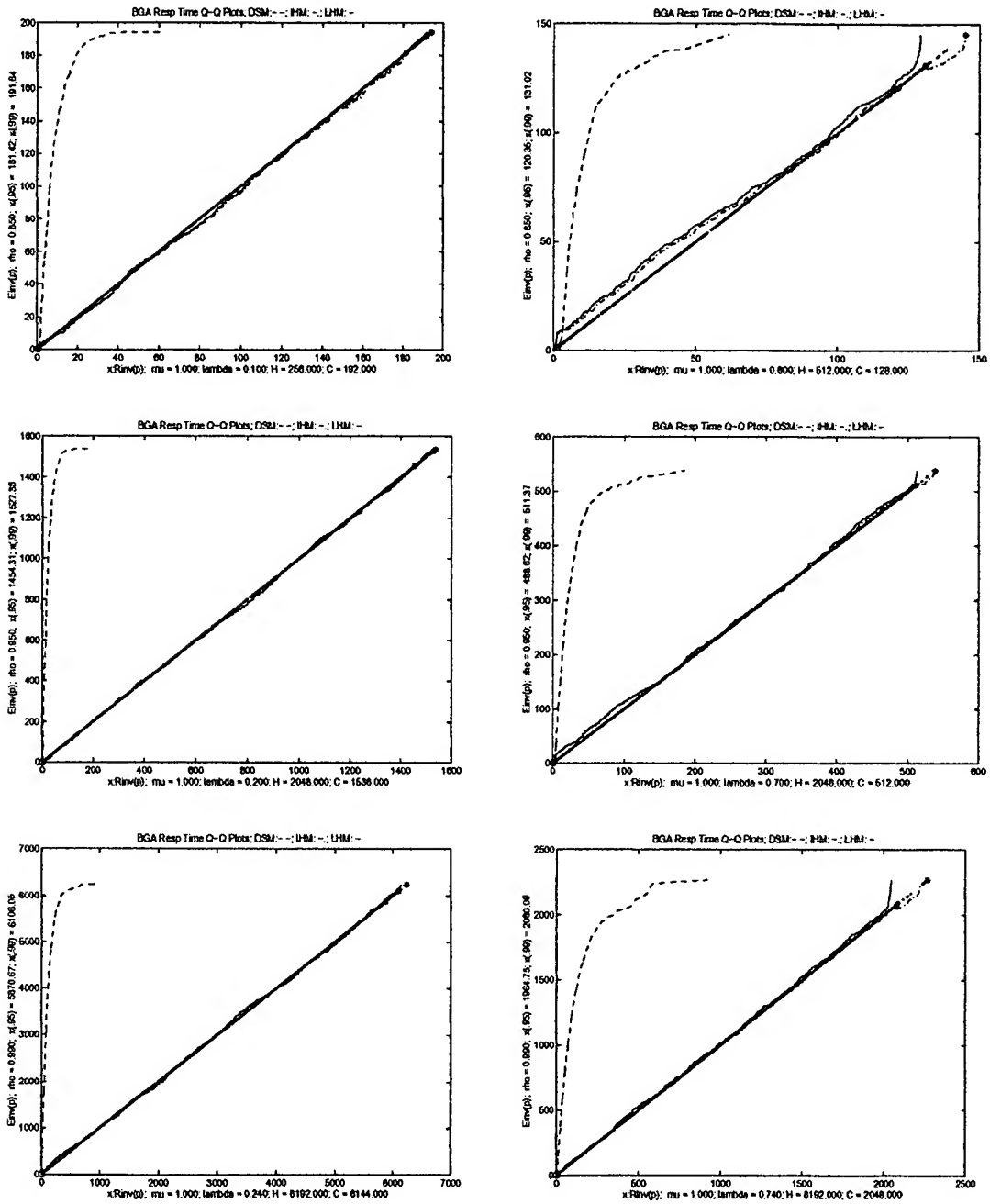


Figure 5.6: BGA LHM Response Time Q-Q Plots

5.2.3 Intermediate Hyperperiods

When neither the SHM nor the LHM are applicable or their application is borderline, we make use of an intermediate hyperperiod model (IHM) (also called a piecewise linear model, PWLM) which collects statistics from various regions of the response time bands. First we give the conditions for which the IHM is reasonable. We then formalize the notion of a response time band structure and illustrate it with several simulation results.

We then identify response time values which tend to both coincide with noticeable slope changes in the response time curves. We call these points *slope change (response time) values* and denote them by V. These values can be related to the response time band structure, and consequently are a function only of the system parameters (i.e. of $\lambda_2, \mu_2 = 1, H$ and C).

The collection of slope change values along with the probability that a response time is less than these values can be used to construct a piecewise linear response time curve. Algorithms to efficiently collect simulation data from process state variables for the probabilities of being less than various critical values are constructed. Our proposed intermediate hyperperiod model lacks in mathematical eloquence, but is efficiently produced and can be rapidly evaluated against observed data.

For the intermediate hyperperiod model, queue lengths grow sufficiently large that the queue discharge will often not occur in one hyperperiod, rendering the LHM inappropriate. Analogously, when the build-up and discharge queue lengths are sufficiently variable among different hyperperiods, the DSM is also inappropriate. This leads us to Conjecture 5.2.5 for the criteria of when the IHM applies.

Conjecture 5.2.5 (BGA IHM Conditions) *For the background aperiodic scheduling model, the IHM is a reasonable estimator of response times when* λ_2 *satisfies*

Equation 5.13.

$$\mu_2(1-\rho_1) - \frac{k_{\rho_1}}{2H} - \frac{1}{2}\sqrt{\frac{k_{\rho_1}^2}{H^2} + \frac{4k_{\rho_1}\mu_2\rho_1}{H}} \leq \lambda_2 \leq \frac{\mu_2(1-\rho_1)^2}{(H\mu_2)^{-1} + (1-\rho_1)}. \tag{5.13}$$

Conjecture 5.13 is an immediate consequence of Conjecture 5.2.1 and Equation 5.8 of Conjecture 5.2.3 (assuming these conjectures are both valid).

Given an arrival time of t mod $H \in [0, H]$, the set of possible response times is defined by a set of bands as shown in Figure 5.7. The empty bands are the result of periodic blocking, where no aperiodic tasks can depart. The non-empty bands are called the *response time bands.* The width of each response time band is $H - C$, consequently the bands on the right hand side of Figure 5.7 (where $C = \frac{3}{4}H$) are much narrower than the bands on the left hand side of Figure 5.7 (where $C = \frac{1}{4}H$). The k^{th} band is labeled B_k. The intermediate hyperperiod model is applicable when response times fall in at least two bands and both bands contain at least a few percent (e.g. $> 3\%$) and the DSM model does not apply.[3] Several observations can be made from Figure 5.7. First, the response time ranges where bands overlap are defined by response time values in ranges $[kH + C, (k+1)H]$ for $k \in \{0, 1, 2, ...\}$. There is no overlap in the ranges $[kH, kH + C]$ for $k \in \{0, 1, 2, ...\}$.

Intuitively, when $H - C$ is small (relative to H), the regions of overlap among any two bands is also small (relative to the area of each band). Let A_k be the area of B_k. Then $A_1 = \frac{1}{2}(H + C)(H - C)$ and $A_k = H(H - C)$ for $k \in, \{2, 3, 4, ...\}$. The area of the overlap region between adjacent bands is $A_o = \frac{1}{2}(H - C)^2$. Let R be a randomly chosen response time. If the $p_k = P[R \in B_k]$ were known, a conservative response time CDF can be constructed using points $(0, 0)$ and $\{(kH + C, p_k^*)\}$, where

[3] In fact, the PWLM works well when estimating response times that are well approximated by the DSM model.

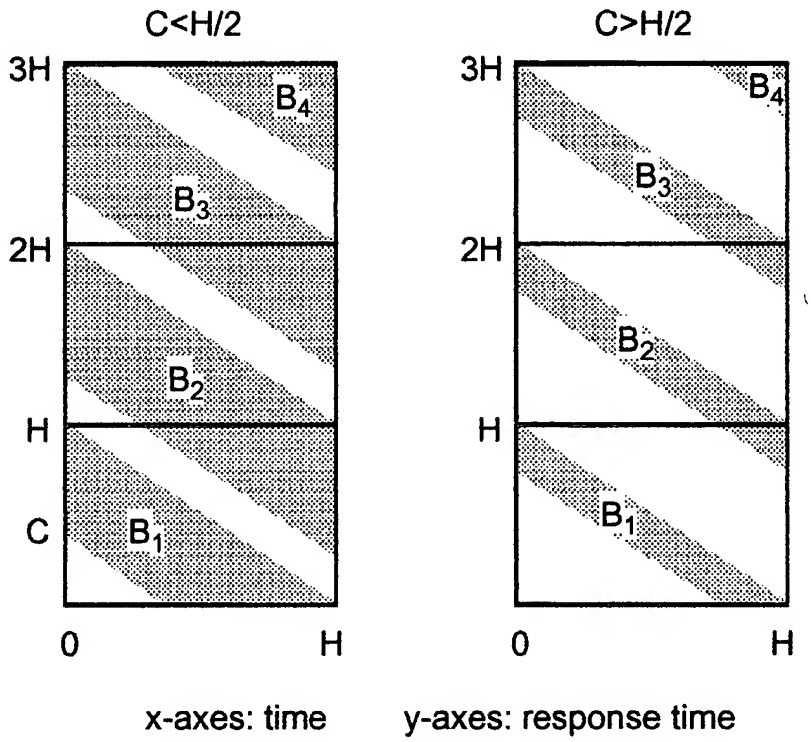


Figure 5.7: BGA IHM Response Time Bands

$p_k^* = \sum_{j=1}^{k} p_j$. When ρ_1, the periodic utilization, is fairly large, this estimator can be reasonable. Other times, a band's overlap regions cannot be attributed solely to the lower portion of the band, and the overlap regions must be accounted individually.

Figure 5.8 shows response time plots as a function of arrival time (within a hyperperiod). It is easy to see the distribution of the data within a hyperperiod is not uniform along the vertical (response time) axis. With the exception of B_1, the density of the samples decreases with an increase in response time. And this density appears approximately uniform across the hyperperiod. In B_1 there is additional structure which we defined in the LHM.

We summarize how the PWLM is constructed:

1. From the response time bands, define the set V of slope change values for

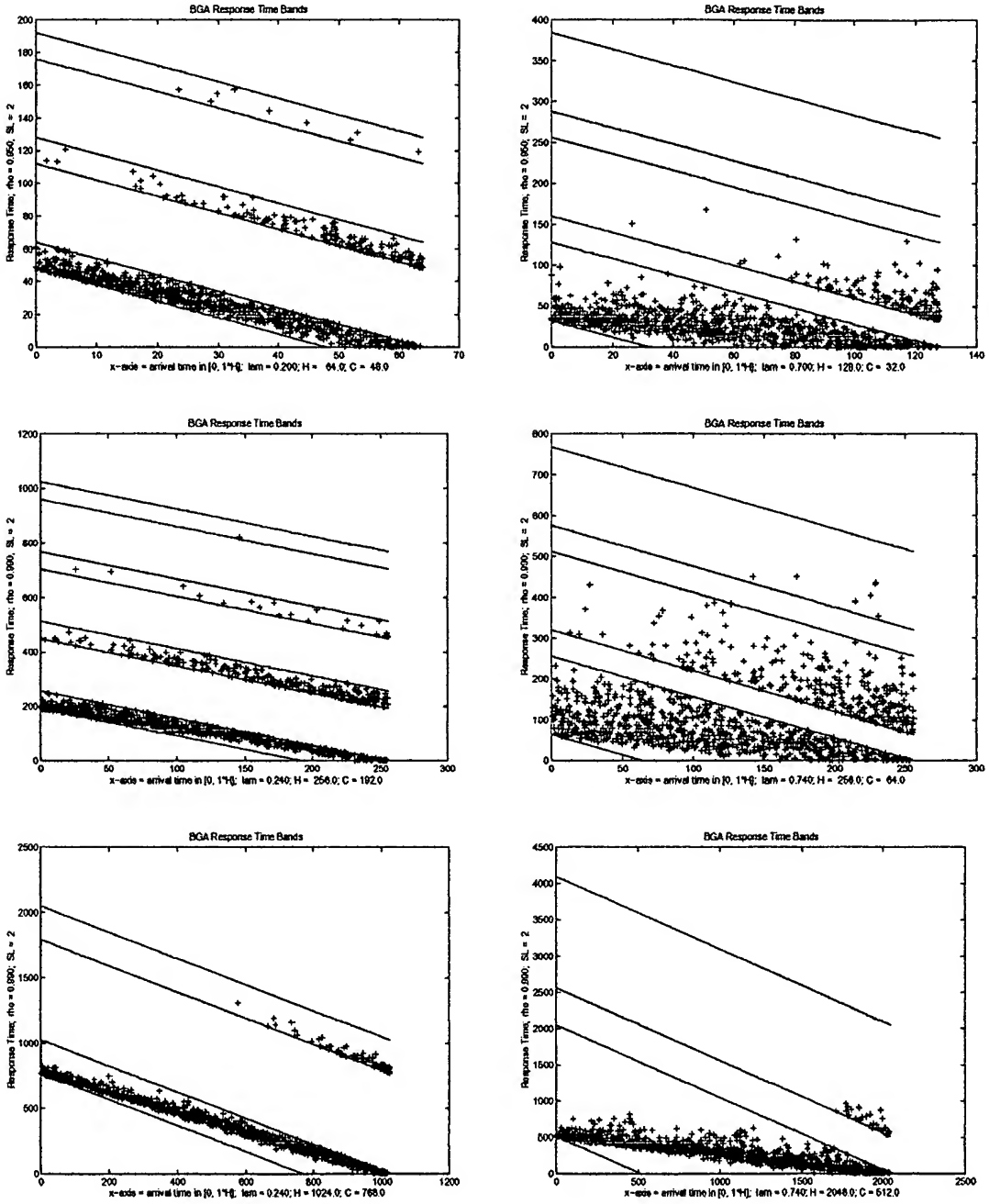


Figure 5.8: BGA IHM Response Time Data Bands

response times. Within B_1, changes occur at $0, x_0, C$, and H.

C is the smallest response time that is resident in both B_1 and B_2 and H is the largest response time that can be resident in both B_1 and B_2.

For each B_k, include points $(k-1)H + C$ and kH. Also include an observed point $v_{\max}$, which is found when we collect values for $p_k^* = P[R \leq v_k]$.

Finally, $V = (0, v_0, v_1, ..., v_{2k-1}, v_{2k}, ..., v_{\max})$, where $v_0 = x_0$, and for $k \geq 1$, $v_{2k-1} = \min((k-1)H + C, v_{\max})$ and $v_{2k} = \min(kH, v_{\max})$. For example, V might be $(0, x_0, C, H, H+C, 2H, 2H+C, 3H, v_{\max})$ where $3H < v_{\max} \leq 3H+C$.

Let l be the index of $v_{\max}$. Vector V now contains $l+2$ points.

2. Next define a set of probability values $P = (0, p_0, p_1, ..., p_l)$ where $p_j = P[v_{j-1} < R \leq v_j]$. Generally the p_j are observed. p_0 and p_1 are special cases, where $p_1 = P[R \leq C]$ and $p_0 = (1 - \mathcal{F}H^{-1})p_1$. $P^* = (0, p_0^*, p_1^*, ..., p_l^* = 1.0)$, where $p_0^* = p_0 = (1 - \mathcal{F}H^{-1})p_1$ and $p_1^* = p_1 - p_0$. Then $p_k^* = \sum_{j=0}^{k} p_j = P[R \leq v_k]$.

3. Apply *lightly populated band clustering* to the bands with the longest response times. If two or fewer percent of the response times lie in a single band, group them with the next lower band. Repeat this process until the band with the largest response times contains at least 3% of the data.

4. If fewer than 98% of the response times are less than $C + x_0$ (which is the LHM), apply *exponential smoothing* to the band containing the largest response times. Exponential smoothing applies except when there is a match with LHM to quantile x(.98).

   In the band for which exponential smoothing applies, the response time distribution characteristics are defined in terms of partial (*vs* full) blocking times. The distribution of blocking times will be covered in Chapter 6.

An algorithm for exponential smoothing follows. Let k satisfy $k \in \{1, ..., l+1\}$ and $p_j \leq 0.02$ for each $j > k$. Let $p^* = p_k^*$ and let x^* be the DSM inverse of p^*, so $x^* = -\log(1-p^*)(\tilde{\mu} - \lambda_2)^{-1}$. Finally, let $x_0^* = v_{k-1}$ and $x_\Delta = x_0^* - x^* \leq 0$. For $x \in [v_{k-1}, v_{\max}]$, define $R_i(x) = 1 - e^{-(\tilde{\mu}-\lambda_2)(x+x_\Delta)}$. In practice, pick some number of evenly spaced v-values for which to compute probabilities.

5. Plot the points defining the PWLM using V as the x-coordinates, and P^* as the y-coordinates. Construct line segments between adjacent points so the PWLM response time EDF is (right) continuous.

The data used by the LPWM can be obtained from system simulation, when one exists. It is often easier and faster to construct a *state variable process* simulation to obtain data points. A state variable simulation algorithm is developed in Section 5.2.4.

Figure 5.9 shows examples of several PWLM response time CDFs overlayed on system simulation EDFs. Figure 5.10 shows corresponding the corresponding response time Q-Q plots.

When system simulation data is used, all of the slope change *points* will fall on the observed empirical distribution function prior to x_0^*, where exponential smoothing begins. Generally the IHM fits reasonably well. It reduces to the LHM when exponential smoothing is not applied, and it is not difficult to see that when there are many bands, it approximates the DSM fairly well. When there are three response time bands, the estimate in the middle band is often conservative, since partial blocking is common within both the second and third bands.

5.2.4 Response Time Variable Process Model

In this section we define the response time behavior in terms of the state variables used to characterize the response time process. The state variable process model is

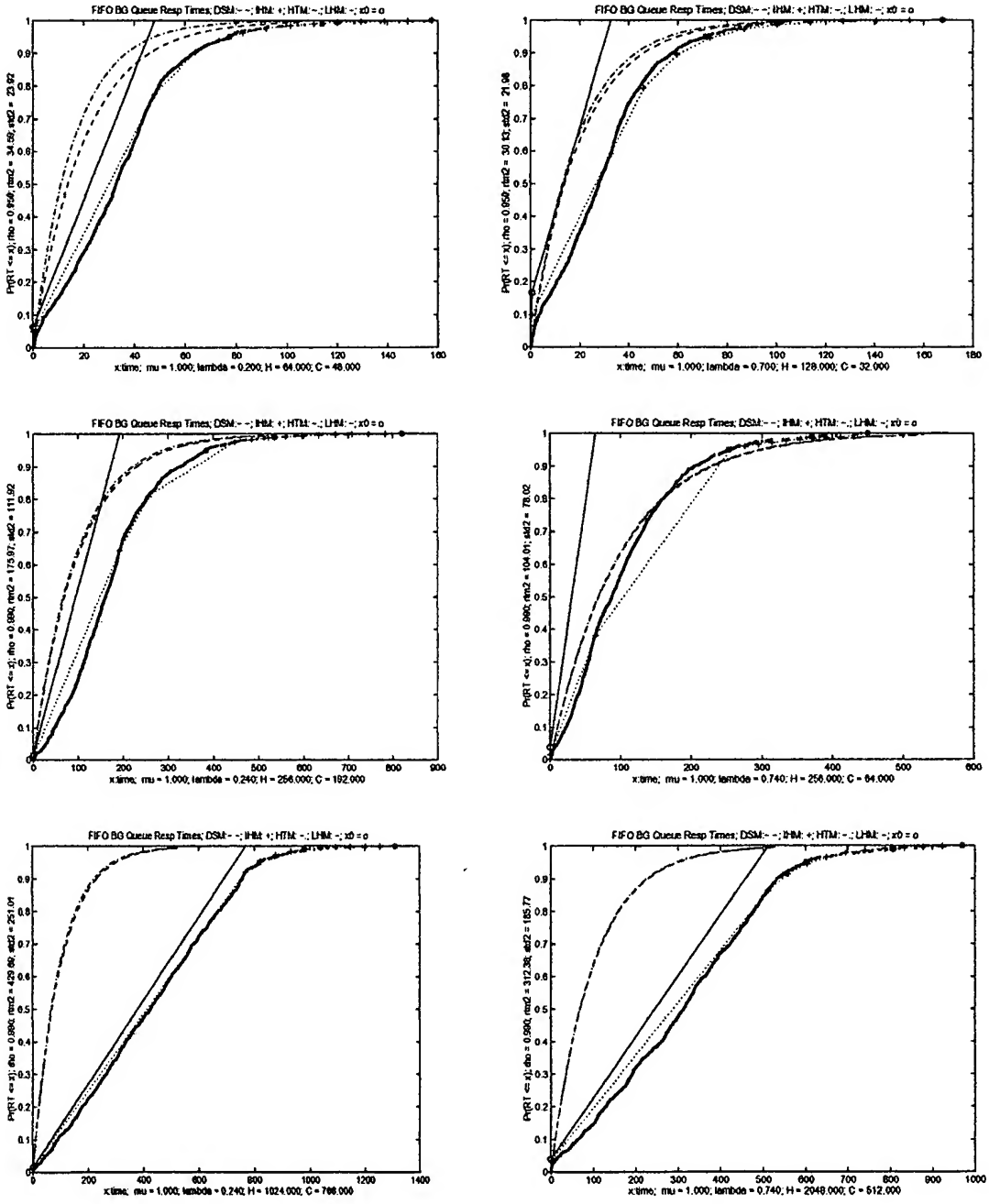


Figure 5.9: BGA IHM Response Time EDFs

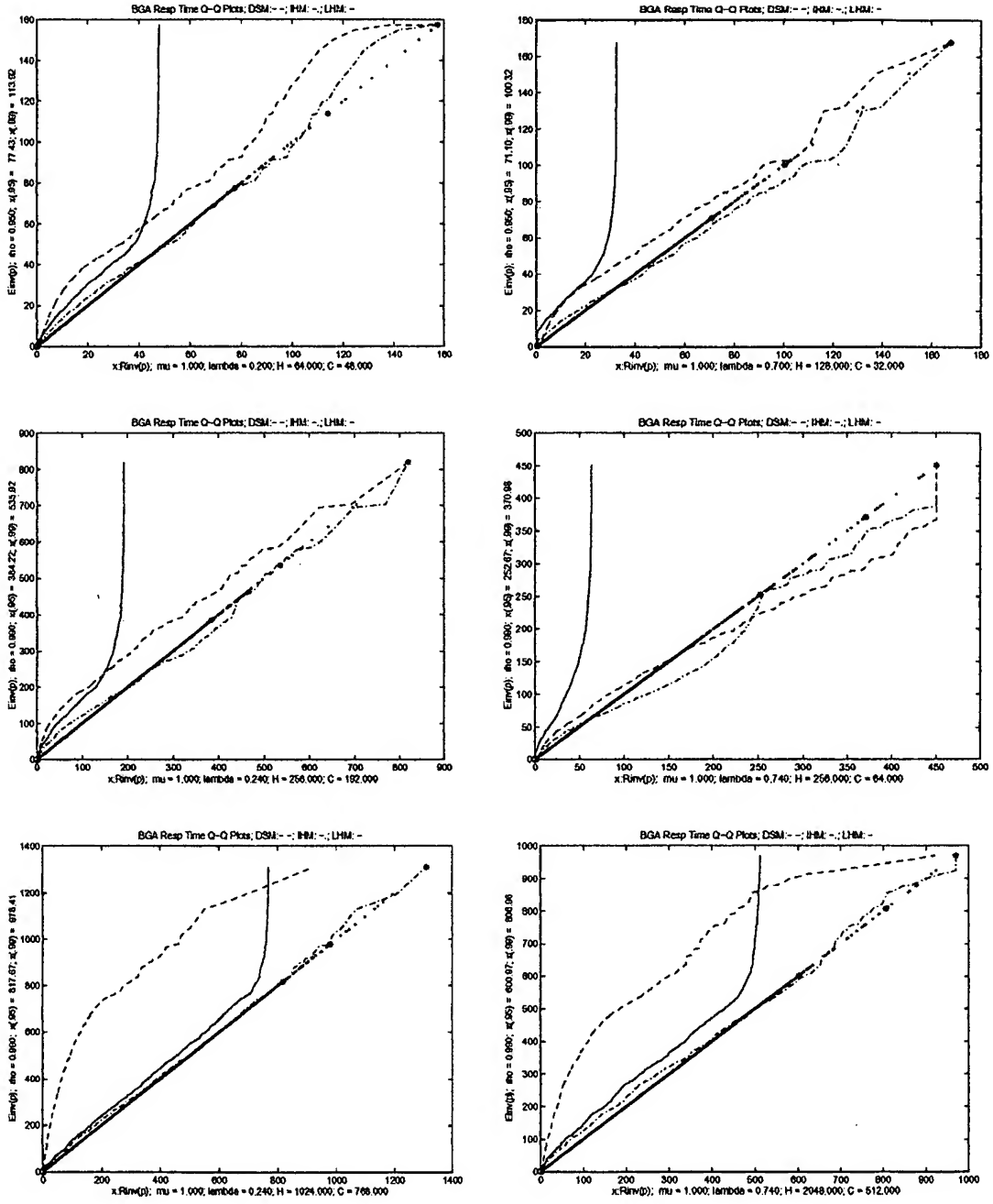


Figure 5.10: BGA IHM Response Time Q-Q Plots

an exact specification of the response time process, although we know of no solutions or non-numeric approximations for its behavior beyond the special cases we have presented.

In this section define $W_a(t)$ be the amount of aperiodic work in the system for an arrival at $t \in [0, H]$ where $W_a(t)$ includes the workload contribution for the newly arriving task. In other words, let $t_k = \sum_{j=1}^{k} T_{2,j}$ = the arrival time of the k^{th} aperiodic to arrive to the system (since time 0). Assume $t_0 = 0 = W_a(0)$, then

$$W_a(t_k) = \max(0, W_a(t_{k-1}) - (t_k - t_{k-1})) + X_{2,k}.$$

Define $R_a(t, W_a(t)) = R_a(t)$ to be the response time of an arrival given the arrival time is at time $t = t \bmod H$ and the aperiodic work in the system just after the instant of arrival is $W_a(t)$. Refering to Figure 5.7, it is not difficult to see that for $k \in \{1, 2, 3, ...\}$,

$$P[R_a(t) \leq kH] = P[W_a(t) \leq k(H - C)] \text{ for } t \in [0, H]. \tag{5.14}$$

Boundaries at multiples of H are nice in that they do not depend on the time of an arrival (modulus H). This is not true for any other set of response time values but some simplifications result for other special cases. In particular, for $k \in \{0, 1, 2, ...\}$, we wish to compute $P[R_a \leq kH + C]$. The set of response time values defined by $\{kH, kH + C \text{ for } k = 0, 1, 2, ...\}$ defines the upper and lower bounds on the points of intersection within the various response time bands. In other words, the range of $B_k = [\max(0, (k-2)H + C), kH]$ and the range of the intersection of $B_k \cap B_{k+1} = [(k-1)H + C, kH]$ and for $k > 1$, the region of B_k that does not overlap with either B_{k-1} or B_{k+1} is $[(k-1)H, (k-1)H + C]$. This partioning captures the band interactions in which response times distributions as a function of time appear similar.

Let $m = \min\{C, H - C\}$ and let $M = \max\{C, H - C\}$. Then Equation 5.15 defines the probability that a newly arriving task at time t mod H falls below $kH+C$ for $k \in \{0, 1, 2, ...\}$.

$$P[R_a(t) \leq kH + C] = \begin{cases} P[W_a(t) \leq k(H - C) + t] & \text{for } t \in [0, m] \\ P[W_a(t) \leq k(H - C) + m] & \text{for } t \in [m, M] \\ P[W_a(t) \leq k(H - C) + (t - m)], & t \in [M, H] \end{cases} \quad (5.15)$$

Equation 5.14 gives us the relationship

$$(k - 1) \leq \frac{W_a(t)}{(H - C)} \leq k,$$

so for

$$k = \lceil \frac{W_a(t)}{(H - C)} \rceil \text{ we have } (k - 1)H \leq R_a \leq kH.$$

Equation 5.15 is then used to decide if $(k - 1)H \leq R_a(t) \leq (k - 1)H + C$ or $(k - 1)H + C \leq R_a(t) \leq KH$. The index for v_j is $j = 2k - 1$ in the former case and $j = 2k$ in the latter case. Figure D.1 (Appendix D) summarizes in pseudo code the algorithm just described to determine the vectors V and P^* as defined in Step 2 of the Algorithm in Section 5.2.3. The algorithm in Figure D.1 is called the state variable process simulation algorithm for BGA IHM estimation.

5.3 Aperiodic System Size Analysis

In this section we consider models for system size, since system size is invariant with respect to the service discipline within a class. The system size analysis proceeds essentially analogously to the response time analysis for short and long hyperperiods, so the presentation is brief. Theoretical means and variances are computed for the

long and short hyperperiod models, and then compared to the data. We leave the exploration of a BGA IHM of system size for future work. Last, we look at the aperiodic queue length at periodic process departure points and compare them to crude theoretical estimations in an attempt to find regions for which asymptotic analysis might work well.

5.3.1 Short Hyperperiods

Using arguments similar to those given in Section 5.2.1 a DSM is used to model aperiodic system size under conditions set forth in Conjecture 5.2.1. The distribution for the limiting system size discrete random variable is given in Equation 5.16.

$$\tilde{N}(n) = P[N_2 \leq n] = 1 - \tilde{\rho}^{(n+1)} = 1 - (\frac{\rho_2}{1-\rho_1})^{(n+1)}. \tag{5.16}$$

By visual inspection, Equation 5.16 tends to be more optimistic and does not fit the state data quite as well as the response time DSM model, but it is certainly a reasonable engineering model. We make no attempt to sharpen the boundary conditions to ensure the applicability of the DSM for system size. Unlike the response time analysis, there is a noticable difference between the DSM and the HTM for system size. In all observed cases, the HTM is more optimistic than the DSM.[4]

Given the applicability of the DSM, the approximate theoretical mean and variance of the response times are those of the approximating models, given in observation 5.3.1.

Observation 5.3.1 (BGA SHM system size mean and variance) *In the background aperiodic scheduling model, when the DSM provides reasonable estimates, the*

[4]In the HTM, it turns out that $N_1 \Rightarrow 0$ so N and $N_2 \Rightarrow$ Equation 5.16. Since periodics do not queue, $|N_1(t) - N(t)| \leq 1$ for all t. No adjustments have been made to the HTM approximations.

approximate theoretical system size mean is

$$m_N = \bar{N} = \tilde{\rho}(1-\tilde{\rho})^{-1},$$

and the approximate system size variance is

$$\sigma_N^2 = \tilde{\rho}(1-\tilde{\rho})^{-2},$$

where $\tilde{\rho} = \lambda_2[\mu_2(1-\rho_1)]^{-1}$. *The DSM model is also an M/M/1 queue, with arrival rate* λ_2 *and service rate* $\tilde{\mu} = \mu_2(1-\rho_1)$. *The system size mean queue length and variance of an M/M/1 queue are given in Table B.1.*

Observe that for fixed ρ_1, *the state data mean and variance do not change with* H. *Unlike the response time DSM, the state mean and variance do change for fixed* H *and varying* ρ_1.

Table 5.5 shows sets of (H, C) values and minimum λ_2 values for which the DSM works. For fixed CH^{-1}, as H increases, so must λ_2 to ensure the queue lengths remain long much of the time. For an equilibrium distribution to exist, $\rho < 1$, so $\lambda_2 = \mu_2\rho_2$ can not be increased without bound, and the DSM is reasonable only for H not too large. In Table 5.5 are observed values for the mean and standard deviation of sample aperiodic system size. There is better agreement for $\rho_1 = 0.75$ and also less variability among observations. $H = 128$ is outside the DSM range for both values of ρ_1.

Figure 5.11 shows several system size EDFs with a theoretical DSM overlay under a range of conditions which meet the criterion in Conjecture 5.2.1. For $\rho = 0.85$ and $\lambda_2 = 0.1$ (not shown) the predictions are fairly optimistic. At the boundaries with equal ρ, the fit is better for $\rho_1 = .25$ than for $\rho_1 = 0.75$ which is not surprising since the aperiodic queue length size will be larger in the former case.

H	C	λ_2	$\hat{m}_n$	$\hat{\sigma}_n$
16	4	≥ 0.693	$m_n = 74$	$\sigma_n = 74.50$
64	16	≥ 0.735	theory $\rho = 0.99$	theory $\rho = 0.99$
16	12	≥ 0.207	$m_n = 24$	and $\sigma_n = 24.50$
64	48	≥ 0.236	theory $\rho = 0.99$	theory $\rho = 0.99$
8	2	0.74	79.71	75.61
32	8	0.74	66.78	62.13
128	32	0.74	90.60	88.46
8	6	0.24	23.49	24.49
32	24	0.24	25.06	22.88
128	96	0.24	30.53	23.51

Table 5.5: BGA SHM System Size Criterion Evaluation

When H becomes long, the DSM becomes much too optimistic.

5.3.2 Long Hyperperiods

For long hyperperiods, the queue length buildup can be analyzed in much the same way as response time delays. First the hyperperiod is decomposed into three regions; a blocking interval $B = [0, C = B]$, a discharge interval $D = [C, \mathcal{F}]$, where $\mathcal{F}$ is the first time the queue length returns to n_0, its value at the beginning of the hyperperiod, and last an M/M/1 interval $M = [\mathcal{F}, H]$. Queue length as a function of hyperperiod is shown in Figure 5.12. For consistency with our value of n_0 in the response time distribution estimators, we take $n_0 = (1-\rho_2)^{-1}\ln((1-\rho_2)^{-1}\rho_1)$.

From Figure 5.12 it is easy to see that for $n \in (n_0, n_0 + \lambda_2 B]$,

$$\Pr(N_1 \leq n) = 1 - \frac{(t_2 - t_1)}{H} = 1 - \frac{1}{\rho_2(\lambda_2 - \mu_2)}[n - (n_0 + \lambda_2 B)]H^{-1}$$

where $t_1 = (n - n_0)\lambda_2^{-1}$ and $t_2 = (n - (\mu_2 B + n_0))(\lambda_2 - \mu_2)^{-1}$ which leads to an

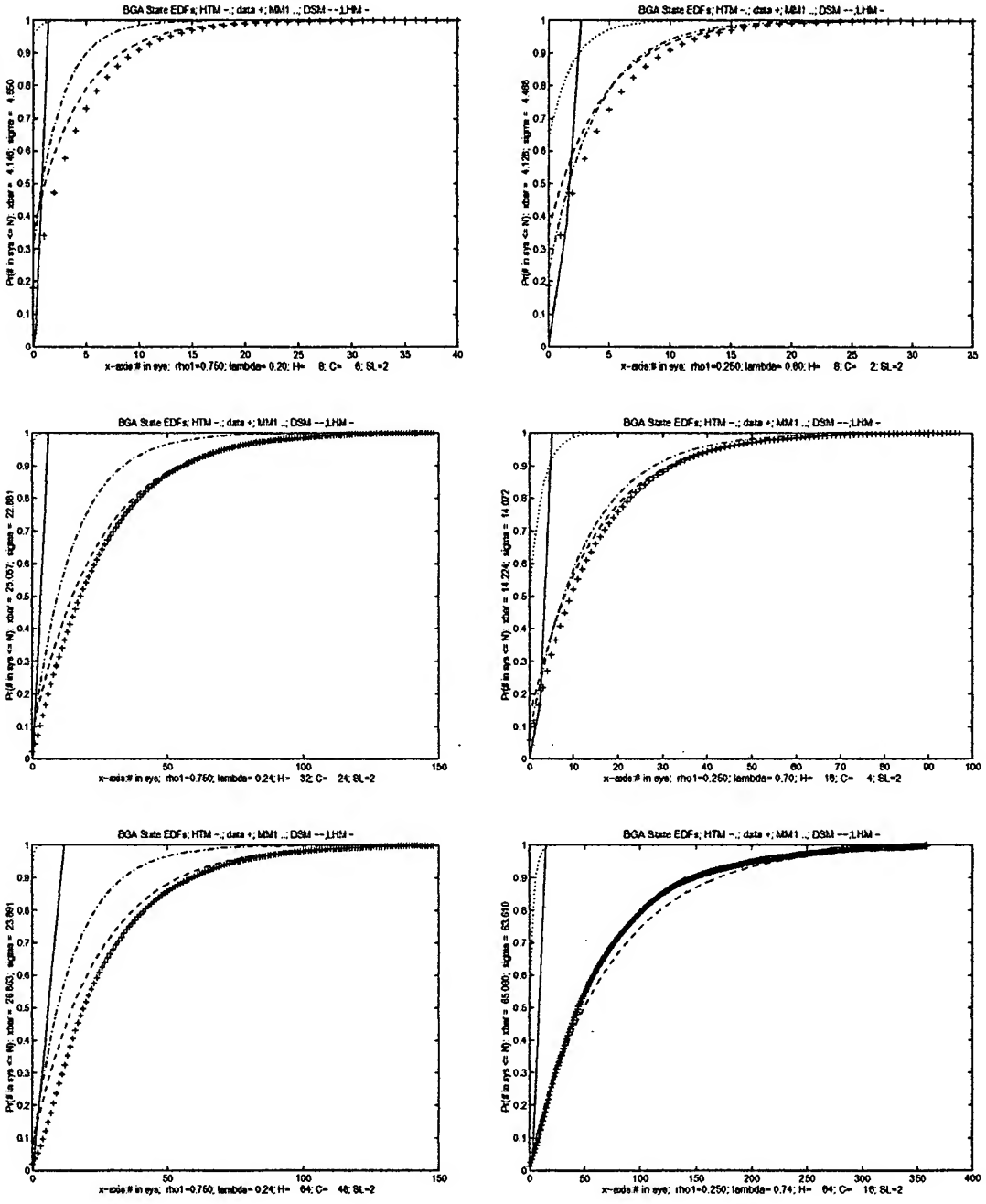


Figure 5.11: BGA SHM System Size EDFs

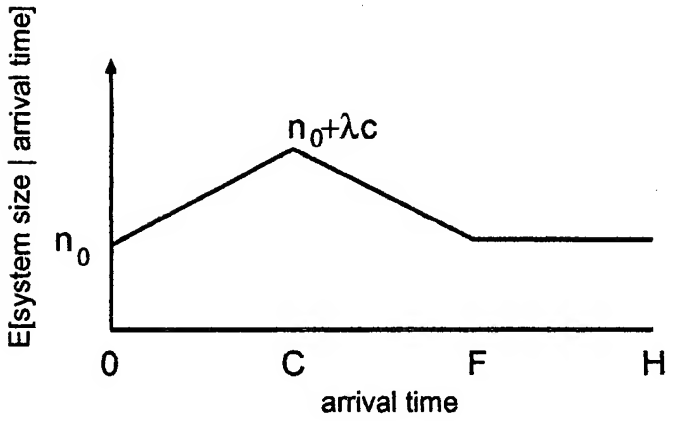


Figure 5.12: BGA LHM: E[System Size | Arrival Time] *vs* Arrival Time

approximating aperiodic system size CDF given in Equation 5.17.

$$\Pr(N_1 \leq n) = \begin{cases} 0 & \text{for } n < 0 \\ 1 - \frac{\mathcal{F}}{H} & \text{for } n = n_0 \\ 1 - \frac{1}{H\rho_2(\mu_2 - \lambda_2)} \cdot (n_0 + \lambda_2 B - n) & \text{for } n \in [n_0, n_0 + \lambda_2 B] \\ 1 & \text{for } n > n_0 + \lambda_2 B. \end{cases} \quad (5.17)$$

Equation 5.17 can be adapted in the obvious ways for values of $n \in [0, n_0]$, but under conditions of heavy traffic, these terms do not significantly change the shape of the distribution. The long hyperperiod model is also appropriate for moderately loaded systems. The CDF for aperiodic system size described by Equation 5.17 is graphed in Figure 5.13.

The conditions for which the LHM appears a reasonable model for system size are characterized in Conjecture 5.2.3. Given applicability of the LHM, the approximate theoretical mean and variance of system size are those of the approximating model, given in observation 5.3.1.

Observation 5.3.2 (BGA LHM system size mean and variance) *In the background aperiodic scheduling model, when the LHM provides reasonable estimates, the*

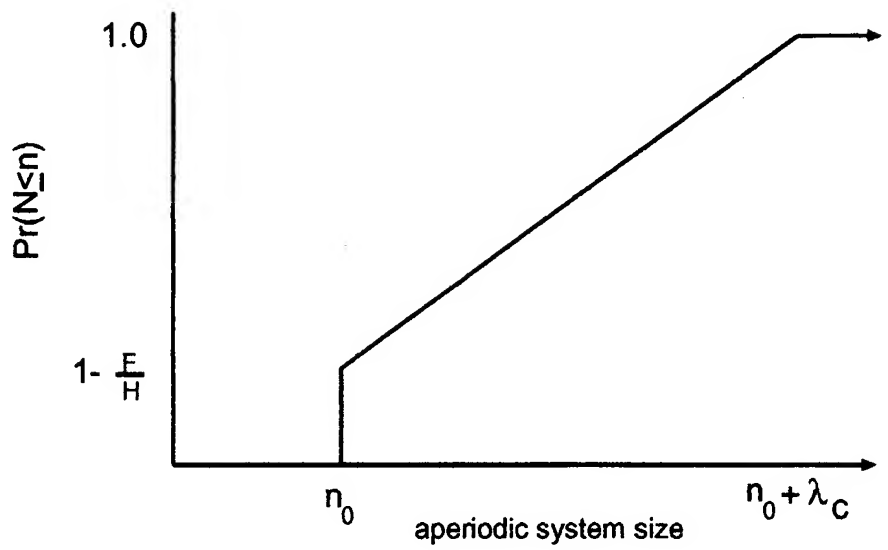


Figure 5.13: BGA LHM Predicted System Size CDF

approximate theoretical system size mean is

$$m_N = \bar{N} = n_0 + \frac{\lambda_2 B \rho_1}{2(1-\rho_2)},$$

and the approximate theoretical system size variance is

$$\sigma_N^2 = \frac{(\lambda_2 B)^2 \rho_1}{1-\rho_2}[\frac{1}{3} - \frac{\rho_1}{4(1-\rho_2)}],$$

both of which follow from standard calculations using Equation 5.17. These calculations have assumed a point mass at n_0, *and continuous (not discrete values) for* $n \in (n_0, n_0 + \lambda_2 B]$. *Note that* $E[N] \approx \lambda_2 E[R]$ *and* $\mathrm{Var}[N] = \lambda_2^2 \mu_2 \mathrm{Var}[R]$.

Table 5.6 shows several examples of theoretical and observed means and standard deviations. The observed values are based on a single simulation run at the respective parameter setting. Even though the means and standard deviations are

H	C	λ_2	$\bar{n}$	$\hat{n}$	σ_n	$\hat{\sigma}_n$
2048	512	0.74	235	235	147	143
4096	1024		434	430	241	242
8192	2048		758	766	463	462
2048	1536	0.24	202	192	112	111
4096	3072		370	373	216	217
8192	6144		753	731	438	433

Table 5.6: BGA LHM System Size Sample Moments

fairly close, the LHM for system size is optimistic in the tails at the boundary conditions. As H increases, the optimism becomes reduced. Figure 5.14 overlays several empirical distribution functions with the system size estimators for long hyperperiod given in Equation 5.17.

5.3.3 Aperiodic Queue Lengths at Periodic Departures

The last topic we look at in this chapter is the distribution of the aperiodic system size at periodic task departure times. Since we don't make use of these results for our background analysis, we make some simple unrefined observations which are summarized in conjecture 5.3.3. However, the distribution of blocking times and the related distribution of aperiodic queue lengths at periodic departure times turns out to be helpful when studying the foreground aperiodic discipline.

Table 5.7 lists the means and standard deviations, when $\rho = 0.99$ of aperiodic queue lengths at both periodic departure times (i.e. when $t \bmod H = C$) and averaged over all time. Let N_b be the number of aperiodics in the system at the time of a periodic departure (just after a blocking period), and let m_b and σ_b denote the mean and standard deviation of N_b, respectively. Several observations can be made. Not

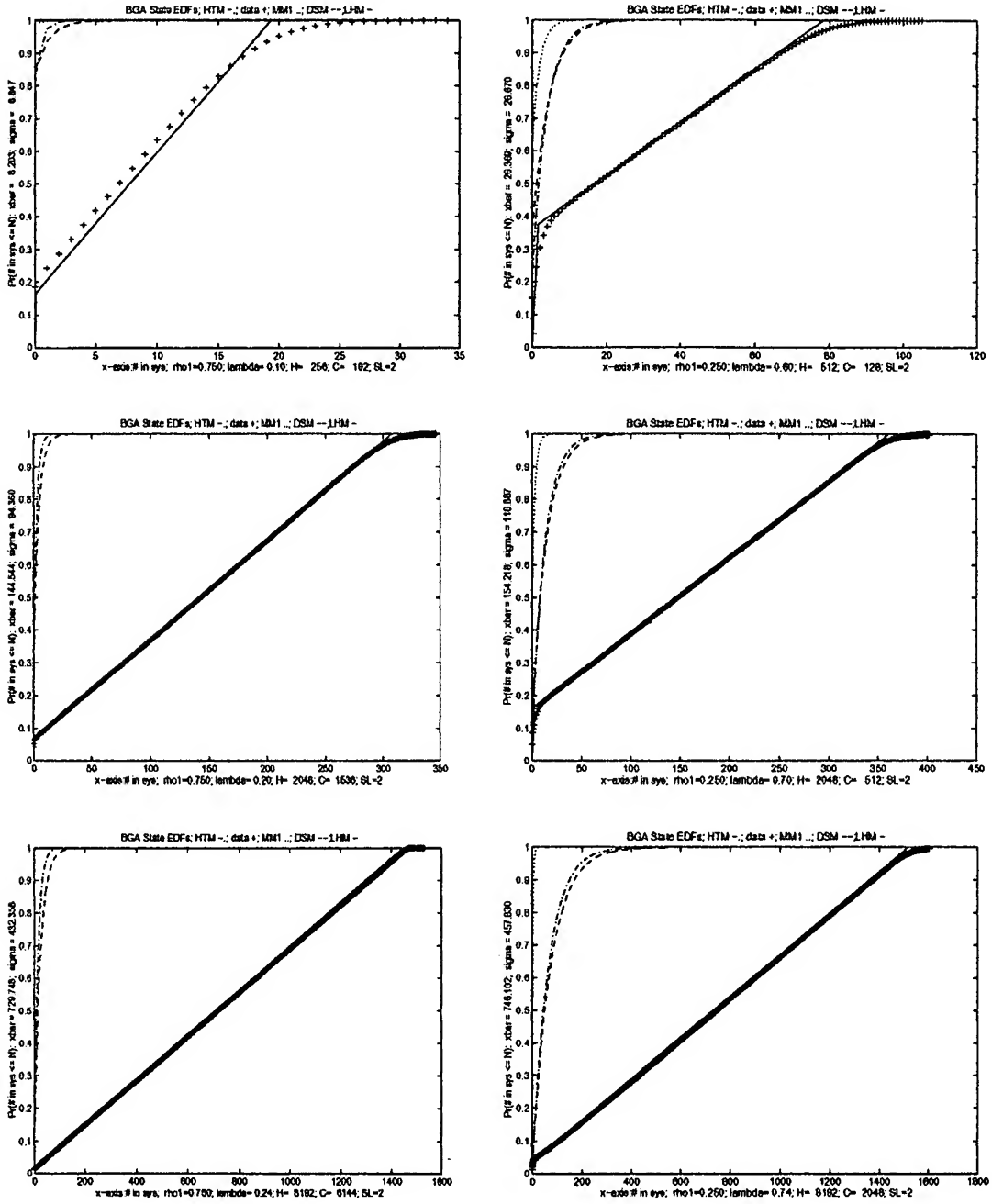


Figure 5.14: BGA LHM System Size EDFs

surprisingly, $\hat{\sigma}_b$ tends to be less than $\hat{\sigma}_n$ (the time averaged sample standard deviation). For $\lambda_2 = 0.74$, $\hat{\sigma}_b$ is essentially constant and does not depend on hyperperiod length in the range considered. Also, when the DSM conditions hold, $\hat{\sigma}_b \approx \hat{\sigma}_n$ and $\hat{m}_b \approx \hat{m}_n$, suggesting that the aperiodic system size behavior at a particular point is representative of all points within a hyperperiod. When either the IHM or LHM hold, $\hat{m}_b > \hat{m}_n$, with the difference increasing with increasing H, again not surprisingly.

H	C	λ_2	$m_{2,b}$	$\hat{m}_b$	$\hat{m}_n$	σ_b	$\hat{\sigma}_b$	$\hat{\sigma}_n$
8	2	0.74	74.00	79.42	78.26	74.50	72.33	75.57
32	8		74.00	75.03	66.78	74.50	67.10	62.13
128	32		IHM	97.57	90.60	74.50	80.72	88.46
512	128		IHM	110.55	108.76	74.50	78.72	84.32
2048	512		378.88	421.71	230.87	74.50	65.33	137.23
4096	1024		757.76	739.43	430.21	74.50	64.12	242.20
8192	2048		1515.52	1543.61	765.82	74.50	68.72	461.85
8	6	0.24	24.00	24.20	23.58	24.50	24.81	24.54
32	24		24.00	27.42	24.78	24.50	24.90	24.66
128	96		IHM	42.90	31.43	24.50	24.55	25.53
512	384		IHM	108.20	61.68	24.50	24.73	36.16
2048	1536		368.64	378.68	192.19	24.50	28.03	110.61
4096	3072		737.28	746.59	373.08	27.15	33.93	217.48
8192	6144		1474.56	1477.43	731.30	38.40	39.07	432.53

Table 5.7: BGA LHM System Size Moments at Departure Times

Let N_d be the average number of aperiodics in the DSM, so $N_d = \lambda_2[\mu_2(1-\rho_1) - \lambda_2]^{-1}$. Define the heavy traffic estimate for the mean number in the system as

$$N_h = \frac{\sigma_n^2}{2m_n} = \frac{\mu_2(1-\rho_1)^2 + \lambda_2}{2(\mu_2(1-\rho_1) - \lambda_2)},$$

where σ_n^2 and m_n are the diffusion coefficients for the system size sketched in Section 3.4.1.

Conjecture 5.3.3 (Aperiodic Queue Length at Periodic Departure Times) *In the background aperiodic scheduling model, under conditions of heavy traffic, the following models approximate the equilibrium distribution of the aperiodic queue length at the time of periodic departures.*

When the DSM is applicable, then

$$Pr[N_{2,b} \leq n] \approx 1 - \tilde{\rho}^{(n+1)}, \text{ for } n \in \{0, 1, 2, ...\} \text{ and } \tilde{\rho} = \frac{\rho_2}{1 - \rho_1}. \tag{5.18}$$

One way to think about Equation 5.18 is that when the blocking time is sufficiently short, fluctuations in queue length are (statistically) indistinguishable at these points. The mean is given by

$$m_b = \frac{\tilde{\rho}}{1 - \tilde{\rho}} = \frac{\lambda_2}{\mu_2(1 - \rho_1) - \lambda_2},$$

and the variance is given by

$$\sigma_b^2 = \frac{\tilde{\rho}}{(1 - \tilde{\rho})^2} = \frac{m_b}{(1 - \tilde{\rho})}.$$

Note that the same consistent bias appears in $\hat{\sigma}_b$ that appeared in $\hat{\sigma}_n$ for the DSM (See Table 5.7).

Under conditions of the LHM

$$\text{As } H \uparrow \infty, \ \sigma_b \to \max(\sigma_d, \sqrt{\lambda_2 C}) \text{ and } m_b \to \lambda_2 C. \tag{5.19}$$

When the quantity $\lambda_2 C$ is large, the accumulation over the blocking interval will provide the dominant terms in the mean and variance. Note, these limiting values are only approximate for $H = 8192$ and $\lambda_2 = 0.24$. Even for $H = 8192$, when $C = \frac{1}{4}H$ the blocking times are not long enough to experience most of the queue length variability.

Based on visual inspection of the data, when in the IHM and LHM regions, if $\hat{\sigma}_b^2 \approx \lambda_2 C$ *then it is plausible that* $N_b \sim \mathcal{N}(\lambda_2 C, \lambda_2 C)$. *When* $\hat{\sigma}_b^2 \approx \sigma_d^2$, *then* $N_b \sim \mathcal{E}(r, \delta)$ *where* $r = N_d^{-1}$ *and* $\delta = \max(m - N_d, 0)$ *where* $m \approx \lambda_2 C$ *(but ususally larger). In the latter case,*

$$P(N_b \leq x) = (1 - e^{-r(x-\delta)})[x \geq \delta].$$

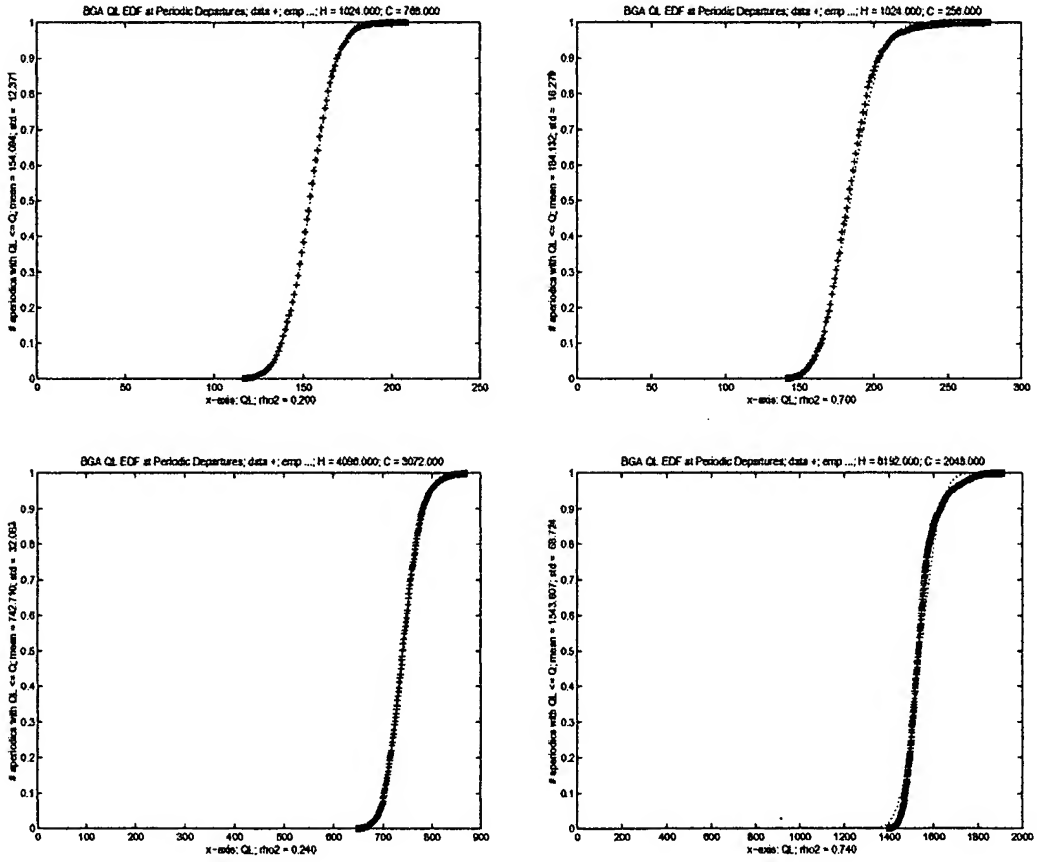


Figure 5.15: BGA: Aperiodic Queue Length EDFs at Periodic Departures

A few EDF plots for aperiodic queue length at the time of periodic departures

under conditions of the LHM are shown in Figure 5.15 for $\rho = 0.99$ and $\rho = 0.95$. For fixed ρ, ρ_1 must be large enough for the asymptotics to apply. When $\rho = 0.95$, the fit is reasonable for both values of ρ_1. When $\rho = 0.99$, the fit in the tails is not good for $\rho_1 = 0.25$. Our conjecture is obviously lacking since even first and second moments are not close for relatively long hyperperiods. It seems possible to more carefully derive estimates of aperiodic queue length distributions at periodic departure instants using heavy traffic theory techniques. However, we do not make use of these results, and we have seen cases where the theoretical estimates are not particularly good in practice.

Chapter 6

Mixed Scheduling: Foreground Aperiodics

Strictly speaking, the title of this chapter is a misnomer since the **periodic task deadline requirement** guarantees every periodic task C units of processor time within H units of its arrival. Periodic tasks will have foreground priority when necessary to meet its deadline requirement. If we define the *slack* to be the time to a periodic task's deadline minus the remainder of its computation time, then aperiodic tasks are only (high priority) foreground tasks when slack is available. When slack becomes unavailable, their priorities drop to the background priority. So "foreground" actually refers to the *possibility* of an aperiodic task having foreground priority for some portion (possibly all) of its system time.

The remainder of this chapter is organized as follows. First, we provide an overview of the system specification while introducing a definition for the periodic blocking time, now a random variable. The periodic blocking time is then conceptually equated to the execution time of a (quasi-)periodic, for which the system anaysis models are developed.

The types of analyses are again subcategorized according to hyperperiod length. This is a simplified categorization, since factors other than hyperperiod length determine the applicability of the various modeling approaches. We again focus on estimation of the right tails of distributions.

6.1 System Specification

The now familiar parameters H, C, λ_2 and μ_2 are sufficient to fully define (but not necessarily analyze) the system behavior.[1] For the foreground aperiodic model, we find it useful to characterize new parameters to help analyze system behavior. In particular, we are interested in estimating the following quantities:

1. the periodic blocking time distribution B (when attainable),
2. the average blocking time $\bar{B}$, when an estimate of the distribution of B is not attainable,
3. ω_0, the probability that blocking does not occur in a hyperperiod, ω_m, the probability that maximum blocking occurs within a hyperperiod (both when an estimate of the distribution of B is not attainable), and
4. $\bar{B}_b$, the average blocking time conditional on some blocking occuring in a hyperperiod, where $\bar{B} = \omega_0 \cdot 0 + (1 - \omega_0) \cdot \bar{B}_b$.

Given $\bar{B}$, a new definition for periodic blocking utilization is $\tilde{\rho}_1 = \bar{B}H^{-1}$. A new definition for system utilization becomes $\tilde{\rho} = \tilde{\rho}_1 + \rho_2$. Note that unless $\bar{B} \approx C$, the aperiodic traffic is often not in a state of heavy traffic. The periodic traffic is sometimes in a state of heavy traffic, but its deadlines are met by design.

6.2 Blocking Time Analysis

A primary difference between the analysis of the foreground and background aperiodic service disciplines is the blocking times in the former are described by a non-constant random variable. At the top of Figure 6.1 is a sequence of hyperperiods illustrating a

[1] These system parameters are defined in Section 5.1.

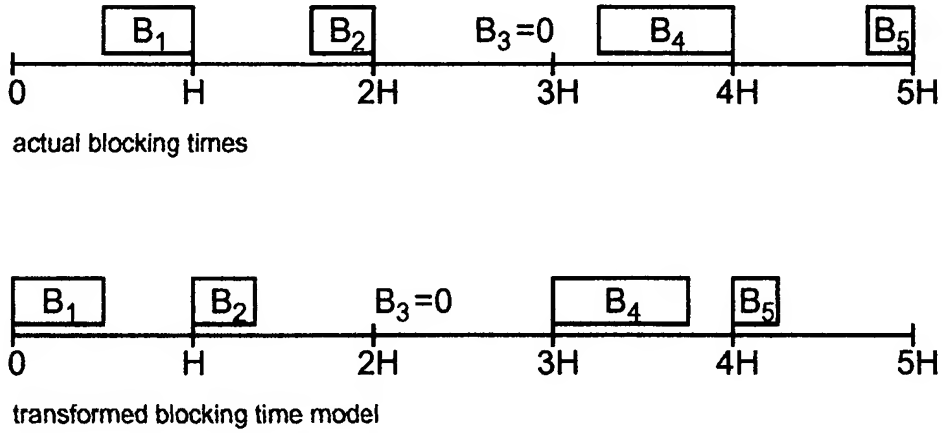


Figure 6.1: FGA Blocking Transformation Example

possible blocking time sequence. Note that the blocking times (when non-zero) occur at the end of the hyperperiod, where aperiodic tasks are suspended to allow the periodic task to complete by its deadline. If we define those hyperperiods in which no blocking occurs to have a blocking time of zero, then the mean time between adjacent blocking times is H. The time between the end of adjacent blocking times is constant and also equal to H. For ease of analysis, we sometimes find it convenient to *assume* that periodic blocking occurs at the start of the hyperperiod (rather than at the end). The bottom of Figure 6.1 illustrates our proposed **transformed aperiodic foreground** model.

The transformed foreground aperiodic scheduling model can now be studied as a background aperiodic scheduling model where the tasks with the periodic arrivals have random (non-constant) execution times defined only by the amount of time for which actual blocking of aperiodic tasks occurs. Assuming the transformed model is a reasonable approximation to the actual model, the blocking time distribution (which conceptually corresponds to the service time distribution for the "high priority" periodic tasks) must be specified.

In the remainder of this section we characterize some general observations about blocking times and propose a candidate model to describe blocking time distributions. Model refinements for different system configurations are described in their respective sections, when determinable.

Let $C(t)$ denote the amount of periodic execution time consumed in $[0, t]$ for $t \in [0, H]$ and $W_2(t)$ denote the aperiodic work queued (or in progress) at time t. Define an *aperiodic idle interval* to be an interval during which there are no aperiodic tasks in the system. The duration of an aperiodic idle interval is measured from the first instant during which there are no aperiodic tasks in the system to the time of arrival of the next aperiodic task. Prior to any blocking within a hyperperiod, the accumulated execution of the periodic task, $C(t)$ is described by the minimum of sum of the aperiodic idle intervals and C.

Depending on the system configuration, one of three things is likely to happen in some percentage of the hyperperiods:

1. No blocking occurs when for all $t \in [0, H]$ the aperiodic workload does not exceed the remaining available time for aperiodics. Strictly speaking, once the periodic task completes no amount of aperiodic workload can cause blocking within the current hyperperiod (but certainly can in subsequent hyperperiods). In symbols, no blocking occurs if $\exists t \in [C, H)$ such that $C(t) = C$.

   Let $\omega_0 = \Pr[\text{no blocking}]$. For H moderately large, $\omega_0 > 0$. For H very large then $\omega_0 \approx 1$, and the aperiodic response time distribution appears (statistically) similar to the idealized M/M/1 response time distribution.

2. Maximum blocking occurs when the system is constantly busy with aperiodic work in the interval $[0, H - C]$. In symbols, maximum blocking occurs when $\forall\ s \in [0, H - C], W_2(s) > 0$. In the case of maximum blocking, the periodic task executes in its entirety at the end of the hyperperiod.

Let ω_m = Pr[maximum blocking]. Only for very short H with heavy traffic is $\omega_m \approx 1$, in which case response times for foreground and background aperiodic services visually appear to be largely indistinguishable. For H moderately long, $\omega_m \approx 0$.

3. Partial blocking occurs when some portion of the execution time of the periodic tasks occurs in at least one aperiodic idle interval and completes in a blocking interval. During the former, no aperiodic tasks are in the system, during the latter at least one aperiodic task is queued. In symbols, $\exists\ s_1, s_2 \in [0, H - C)$ such that $0 \leq s_1 < s_2 < H - C$, and $\forall s \in [s_1, s_2], W_2(s) = 0$. Also, $\exists\ t \in [H - C, H)$ such that $W_2(t) > (H - t) - (C - C(t))$ is satisfied. The interval in the first condition requires that the aperiodic idle interval have non-zero measure.

   Let ω_p = Pr[partial blocking].

   Let $B_p(x)$ = Pr[partial blocking time is $\leq x \mid$ some blocking occurred] and let b_p be the corresponding pdf for B_p.

Equation 6.1 summarizes the blocking time notation.

$$
\begin{array}{ll}
C(t) & \text{the accumulated period compute time at } t \in [0, H], 0 \leq C(t) \leq C \\
W_2(t) & \text{the amount of aperiodic work in the system at } t \in [0, H] \\
n_a(t) & \text{the number of aperiodic idle intervals in } [0, t], t \in [0, H] \\
\omega_0 & \text{the probability of no blocking in a hyperperiod} \\
\omega_m & \text{the probability of maximum blocking in a hyperperiod} \\
\omega_p & \text{the probability of partial blocking} \\
B_p(x) & \Pr[\text{blocking time } \leq x \mid \text{partial blocking}] \\
b_p(x) & dB_p(x)/dx \\
B(x) & \Pr[\text{blocking time } \leq x] \\
b(x) & dB(x)/dx
\end{array}
\tag{6.1}
$$

Equation 6.2 describes $B(x)$, the probability that the blocking time does not exceed x in an arbitrary hyperperiod.

$$
B(x) = \begin{cases}
0 & \text{for } x < 0 \\
\omega_0 & \text{for } x = 0 \\
\omega_0 + \omega_p B_p(x) & \text{for } 0 < x < C \\
1 & \text{for } x \geq C
\end{cases}
\tag{6.2}
$$

Denote the associated pdf for $B(x)$ by $b(x)$ which is shown in Equation 6.3.

$$
b(x) = \begin{cases}
\omega_0 & \text{for } x = 0 \\
\omega_p b_p(x) & \text{for } 0 < x < C \\
\omega_m & \text{for } x = C \\
0 & \text{otherwise}
\end{cases}
\tag{6.3}
$$

Note that

$$\bar{B} = \omega_0 0 + \omega_p \bar{B}_p + \omega_m C = \omega_p \bar{B}_p + \omega_m C.$$

We currently collect ω_0, ω_m and often $\bar{B}_p$ from data generated by the system simulation.

6.2.1 Estimating $\omega_0 = \Pr[\text{no blocking}]$

In this section we give two mathematical expressions for asymptotic approximations of ω_0, the probability that no blocking occurs in a hyperperiod. Unfortunately, neither approximation turns out to provide good estimates so we ultimately rely on simulation data for these parameter values. One shortcoming of our approximations is that they consider the behavior of a single hyperperiod and do not take into account the feed forward nature of aperiodic backlogs from hyperperiods with blocking. Nonetheless it is informative to observe the size and patterns of discrepancies between observed and predicted values.

For the $M/M/1$ (and $M/G/1$) queue, the distribution of each idle interval is $\mathcal{E}(\lambda_2)$. Given the number of aperiodic idle intervals in $[0,t]$ is $n_a(t)$, then $C(t) \approx \mathcal{G}_H(\lambda_2, n_a(t))$, when $C(t) < C$.[2] Let t_b be the first instant at which periodic blocking occurs in the hyperperiod. If no blocking occurs, then $t_b = H$.

The definition of an aperiodic idle interval at the start and completion of hyperperiods adds complications. For simplicity, if at the start (end) of the hyperperiod there are no aperiodics, we simply assume this begins (ends) an aperiodic idle interval, even though the actual beginning (ending) almost certainly started (ended) in the previous (next) hyperperiod. For very long hyperperiods, two fractional aperiodic idle intervals is but a small percentage of the hundreds or thousands that occur.

[2] $\mathcal{G}_H$ is a gamma distribution conditioned on the value being less than H. The observed aperiodic idle time cannot exceed the observation period.

Using the CLT for renewal processes (see the example of Lemma C.1.1), for $t < t_b$ with t_b large, as $t \uparrow t_b$,

$$n_a(t) \sim \mathcal{N}(t\lambda_2(1-\rho_2), t\lambda_2(1-3\rho_2+4\rho_2^2)). \tag{6.4}$$

If the estimates of the number of aperiodic idle intervals is poor, then our Gamma approximation for the aperiodic idle time is also likely to be poor. One would not expect an asymptotic result to work well for short or even moderate H.

Our first estimate for ω_0 is given in Equation 6.5. When we approximate ν by a normal random variable and we do not integrate over domains where $\nu < 0$.

$$\begin{aligned} \omega_{0,1} = \omega_0 &\approx \textstyle\sum_{\nu=0}^{\infty} \Pr[G_{\lambda_2,\nu} \geq C | G_{\lambda_2,\nu} \leq H] \Pr[n_a(H) = \nu] \\ &\approx \textstyle\int_0^{\infty} \Pr[G_{\lambda_2,\nu} \geq C | G_{\lambda_2,\nu} \leq H] d\Pr[n_a(H) \leq \nu]. \end{aligned} \tag{6.5}$$

For long hyperperiods, we consider a second estimate of ω_0. Let I_j be the length of the j^{th} aperiodic idle interval. Then $E[I_j] = \lambda_2^{-1}$ for each j and by the strong law of large numbers (SLLN), $\lim_{n\to\infty} n^{-1} \sum_{j=1}^{n} I_j = \lambda_2^{-1}$ w.p.1. Let $n_a^+ = \max(0, n_a)$ be an estimate of the number of aperiodic idle intervals in the hyperperiod where $n_a = n_a(H) \sim \mathcal{N}(H\lambda_2(1-\rho_2), H\lambda_2(1-3\rho_2+4\rho_2^2))$. For H large, $n_a^+ \approx n_a$ (with high probability). Our second estimate of ω_0 is given in Equation 6.6.

$$\omega_{0,2} = \omega_0 \approx \Pr(n_a(H) \geq \lambda_2 C) = \Pr(Z \geq \frac{[\lambda_2 C - H\lambda_2(1-\rho_2)]}{[H\lambda_2(1-3\rho_2+4\rho_2^2)]^{\frac{1}{2}}}), \tag{6.6}$$

where $Z \sim \mathcal{N}(0,1)$. For short hyperperiods, when not using n_a^+ Equation 6.6 is essentially assigning positive measure to cases with an approximated negative number of aperiodic idle intervals resulting in an erroneously high probability for no blocking.

For $t \geq t_b$, the distribution of $C(t) \not\sim \mathcal{G}_H(\lambda_2, n_a(t))$. The complete description

H	C	ρ	λ_2	$m_{n_a(H)}$	$\sigma_{n_a(H)}$	ω_0	$\omega_{0,1}$	$\omega_{0,2}$
8	2	0.99	0.74	1.54	2.40	0.022	0.433	0.510
32	8			6.16	4.79	0.066	0.482	0.520
128	32			24.63	9.59	0.145	0.516	0.539
512	128			98.51	19.17	0.261	0.560	0.578
2048	512			394.04	38.35	0.446	0.632	0.654
4096	1024			788.07	54.23	0.516	0.687	0.712
8192	2048			1576.14	76.70	0.727	0.756	0.785
8	6		0.24	1.46	0.99	0.068	0.187	0.508
32	24			5.84	1.98	0.096	0.327	0.515
128	96			23.35	3.96	0.233	0.454	0.531
512	384			93.39	7.92	0.380	0.523	0.562
2048	1536			373.56	15.84	0.639	0.571	0.622
4096	3072			747.11	22.40	0.731	0.604	0.670
8192	6144			1494.22	31.68	0.902	0.649	0.733
8	2	0.95	0.70	1.68	2.19	0.167	0.464	0.551
32	8			6.72	4.39	0.301	0.551	0.601
128	32			26.88	8.78	0.512	0.655	0.695
512	128			107.52	17.56	0.785	0.808	0.846
1024	256			215.04	24.83	0.875	0.893	0.926
2048	512			430.08	35.11	0.980	0.961	0.979
8	6		0.20	1.28	0.95	0.327	0.181	0.534
32	24			5.12	1.89	0.476	0.334	0.567
128	96			20.48	3.79	0.741	0.504	0.632
512	384			81.92	7.57	0.938	0.652	0.751
1024	768			163.84	10.71	0.988	0.724	0.831
2048	1536			327.68	15.15	1.000	0.805	0.912
8	2	0.85	0.60	1.92	1.75	0.406	0.541	0.659
32	8			7.68	3.51	0.649	0.719	0.794
64	16			15.36	4.96	0.809	0.815	0.877
128	32			30.72	7.01	0.902	0.907	0.950
512	64			122.88	14.02	1.000	0.995	0.999
8	6		0.10	0.72	0.77	0.739	0.141	0.562
32	24			2.88	1.54	0.912	0.293	0.622
64	48			5.76	2.18	0.978	0.398	0.670
128	96			11.52	3.08	0.996	0.517	0.734
512	384			46.08	6.16	1.000	0.769	0.894

Table 6.1: Two estimates and experimental data for ω_0

of the distribution of $C(t)$ is given in Equation 6.7.

$$C(t) \sim \begin{cases} \mathcal{D}(0) & n_a(t) = 0 \text{ and } t \leq H - C \\ \mathcal{G}_H(\lambda_2, n_a(t)) & \text{for } 0 \leq t \leq t_b, n_a(t) > 0 \text{ and } C(t) < C \\ \mathcal{G}_H(\lambda_2, n_a(t_b)) + (t - t_b) & \text{for } t_b \leq t \leq H, n_a(t) > 0 \text{ and } C(t) < C \\ (t - (H - C)) & \text{for } n_a(t) = 0 \text{ and } t > H - C \\ \mathcal{D}(C) & \text{otherwise} \end{cases} \tag{6.7}$$

When $t_b \ll H$, the idle counts are not approximately normally distributed and the accumulated aperiodic idle times are not approximately gamma distributed which is likely one reason why our estimates are quite poor for short hyperperiods.

However, Table 6.1 shows neither estimate of ω_0 is particularly good for large values of H either. Values in the column labelled $\omega_{0,1}$ were calculated using Equation 6.5 and values in the column labelled $\omega_{0,2}$ were calculated using Equation 6.6. Only for long hyperperiods when $\rho_1 = 0.25$ (i.e. when most of the traffic is aperiodic), are the estimates even approximate to the data and to each other. Our estimates are pessimistic when $\rho_1 = 0.75$ for the longer hyperperiods.

Let ω_{0,ρ_1} be the value of ω_0 at ρ_1 (for some fixed ρ and H). In practice we consistently *observed* that for equal utilizations and all H, $\omega_{0,\rho_1=0.75} \leq \omega_{0,\rho_1=0.25}$. In contrast, our predicted estimates consistently estimated $\omega_{0,\rho_1=0.25} \leq \omega_{0,\rho_1=0.75}$. Despite this troublesome inconsistency, we strongly suspect the *observed* inequality generalizes.

Informally, fix ρ and consider only the aperiodic traffic. Prior to periodic blocking, in the absense of any initial aperiodic backlog, $E[C(t)] = (1 - \rho_2)t$. For $s \geq \rho_1(1-\rho_2)^{-1}H$, $E[s] = C$. Now fix ρ_2 and H, then $s \uparrow$ as $\rho \uparrow$, suggesting increased probability of blocking at the end of the hyperperiod for heavily utilized servers. For fixed ρ and ρ_2, $s \uparrow$ as $H \uparrow$ suggesting more aperiodic idle intervals per hyperperiod tend to occur, in turn suggesting that fewer aperiodic tasks will experience blocking. Lastly, for fixed H and ρ, as $\rho_2 \uparrow$ then $s \downarrow$ suggesting that the periodic task tends to complete earlier in the hyperperiod. This in turn suggests the probability of blocking decreases with decreasing aperiodic traffic, which is again consistent with our observations.

Since we were unsuccessful at finding an explicit expression, we instead rely on simulation data for values of ω_0. We close this section with Conjecture 6.2.1, which we later use.

Conjecture 6.2.1 (As hyperperiods increase blocking decreases.) *Under the foreground aperiodic scheduling model, when ρ and ρ_1 are fixed, then $\omega_0 \uparrow 1$ as $H \uparrow \infty$.*

Using the informal reasoning in the previous paragraphs, in the absence of aperiodic backlog and of periodic blocking, the expected time for aperiodic idle time accumulation to (first) equal C is

$$s = \frac{\rho_1}{1-\rho_2} H,$$

which clearly increases with H when ρ and ρ_1 are fixed. Note, when $H = \infty$, the FGA discipline reduces to the M/M/1 queue and $\omega_0 = 1$ for all $\rho \leq 1$. Also, all data and all our various attempts at calculating ω_0 increase monotonically with H for fixed ρ and ρ_1.

6.2.2 Estimating ω_m = Pr[max blocking]

Maximum blocking occurs when the system never empties of aperiodic tasks in $[0, H-C]$. If at time 0, an aperiodic task arrived to an otherwise empty aperiodic queue, then ω_m would be the probability of a busy interval exceeding $H-C$. For the M/M/1 queue, the density of the busy interval is given in Equation 6.8 ([35], pg. 33).

$$\mathcal{B}'(x) = \mu e^{(\lambda+\mu)x}[I_0(2\sqrt{(\lambda\mu)}x) - I_2(2\sqrt{(\lambda\mu)}x)] \; [x > 0], \tag{6.8}$$

where $I_r(x)$ is the modified Bessel function of order r defined by

$$I_r(x) = \sum_{j=0}^{\infty} \frac{(x/2)^{r+2j}}{j!(r+j)!}. \tag{6.9}$$

The integral, $\mathcal{B}(x) = \int_0^x \mathcal{B}'(y)\, dy$ can be evaluated with the aid of a computer. However $\omega_m \neq (1 - \mathcal{B}(H-C))$, since multiple aperiodics queued at the beginning of the

hyperperiod gives rise to multiple busy periods (calculated using convolutions of the density given in Equation 6.8) which becomes more complex to evaluate both analytically and computationally. The analytic challenge is to describe the distribution of queued aperiodics at the end of the hyperperiod, which is related to the blocking distributions (since during blocking times, queue lengths only grow). We will later look at some sample aperiodic queue length distributions at periodic departure times.

If we were to consider more general service time distributions for the aperiodic tasks, the busy period distribution has also been found for the M/G/1 queue ([35], pg 58). It involves summing an infinite number of k-fold convolutions, where k is the index of summation. This gives rise to a problem of computation and approximation. So we rely on simulation data for values for ω_m, just as we did for ω_0.

When the observed value of $\omega_m \approx 1$, the foreground and background aperiodic scheduling models appear essentially the same. When the observed value of $\omega_m > 0$ and $\omega_m \not\approx 1$, the blocking time distribution has a point mass of ω_m at the value C. Examples are given in subsequent sections.

6.2.3 Estimating ω_p = Pr[partial blocking] and B_p

The relationship $\omega_0 + \omega_p + \omega_m = 1$ clearly holds, so if ω_0 and ω_m were computable, then so would be ω_p. Several weighting distributions were observed among the ω's.

1. $\omega_m \approx 1$: In this case $\omega_0 \approx 0$ and $\omega_p \approx 1 - \omega_m$. For simplicity, the blocking time distribution is assumed to be $\mathcal{D}(\bar{B}) \approx \mathcal{D}(C)$.

2. $\omega_m \approx 0$: When there is no maximum blocking, there may not be, but typically is some partial blocking.

   (a) When $\omega_p > 0$, the blocking time distribution is reasonably approximated by an exponential with parameters determined from either heavy traffic

theory or the degraded server model. This will be explored in Section 6.6.

(b) When $\omega_0 \approx 1$, the blocking time distribution is relatively unimportant since it has so little weight.

3. $\omega_m, \omega_p, \omega_0 > 0$: When $\omega_m > 0$, there are tasks with completion times that experience at least one full block, and possibly more. For those tasks, a degraded server model is appropriate. For tasks with completion times experiencing no blocking, the M/M/1 model is appropriate. For this case we model task response times as a mixture of response time models for the M/M/1 and DSM queues and find cases where the mixture model works well, but also also find cases where including the blocking time distribution (perhaps as a part of a completion time distribution) might lead to improved estimates which we defer to future work.

6.2.4 Task *vs* Hyperperiod Blocking Probabilities

The previous sections focused on the probability of no blocking in a hyperperiod which is different from the probability an individual aperiodic task experiences blocking. Let $\tilde{\omega}_0$ be the probability that an aperiodic task experiences no delays due to blocking. An aperiodic task, α, is said to experience blocking delays when, while waiting or in service, there is some aperiodic task with a completion time that includes some portion of a periodic task's execution time. In other words, an aperiodic task's response time includes no blocking only if all of its waiting time is due to aperiodic processing and its completion time is equal to its execution time.

When $\omega_m = 0$, we use the task blocking approximation given in Equation 6.10.

$$\tilde{\omega}_0 = \omega_0 + (1 - \omega_0)(1 - \bar{F}_b H^{-1}) = 1 - \frac{\bar{F}}{H} = 1 - \frac{\tilde{\rho}_1}{(1 - \rho_2)}. \tag{6.10}$$

Note the similarity to Equation 5.9 when $x = x_0$ which models the probability an

aperiodic task experiences no blocking in the BGA LHM. The assumption that $\omega_m = 0$ says that tasks do not queue for multiple hyperperiods and by implication will typically depart within H time units of their arrival. Now, let γ be the arrival time of an aperiodic task and consider the interval $[\gamma, \gamma + H]$. By the PASTA property, an arbitrary aperiodic arrival will see time averages, and hence experience no blocking under two conditions. First the task might arrive to a hyperperiod interval $[\gamma, \gamma + H]$ in which no blocking occurs which happens with probability ω_0. Also by the PASTA property, the interval $[\gamma, \gamma + H]$ contains partial blocking with probability $1 - \omega_0$. In this case, the task must arrive and depart in the M/M/1 part of the blocking hyperperiod, which happens with approximate probability $(1-\omega_0)(1-\bar{F}_b H^{-1})$, where $\bar{F}_b = \mu_2 \bar{B}_b (\mu_2 - \lambda_2)^{-1} = \bar{B}_b (1 - \rho_2)^{-1}$ is the expected first return time to the M/M/1 state when in a hyperperiod with blocking. (See Section 5.2.2 for the derivation of $\bar{F} = \bar{B}(1-\rho_2)^{-1}$ and hence for $\bar{F}_b$.) Note the similarity of form in Equation 6.10 with the equation for ω_0 given in Section 5.2.2.

Since we have assumed $\omega_m = 0$, $\bar{B}_b = \bar{B}_p$. However, since our definition of $\tilde{\rho}_1$ does not depend on partial blocking (i.e. on $\omega_m = 0$), we use Equation refeq:task-blk to describe task blocking for all of our proposed FGA models. Note that when $\tilde{\rho}_1 = \rho_1$, our definition of task blocking agrees with the derived definition for the BGA LHM case.

6.3 FGA System Model Parameters

This section contains a notation summary of variables (parameters and/or statistics) used for the analysis of foreground aperiodic (FGA) scheduling *assuming* the transformed model illustrated in Figure 6.1 is a reasonable approximation. Some of the

notation has been previously defined, and some has yet to be defined.

$$
\begin{aligned}
B_k &= \text{blocking time in hyperperiod } k, 0 \leq B_k \leq C \\
\omega_0 &= \text{fraction of hyperperiods for which } B_k = 0 \\
\omega_b &= 1 - \omega_0 \\
\bar{B} &= \text{average of all } B_k \\
\tilde{\rho}_1 &= \text{effective periodic utilization} = \bar{B}H^{-1} \\
\tilde{\rho} &= \text{effective utilization}, = \tilde{\rho}_1 + \rho_2 \\
\bar{B}_b &= \text{average blocking time over all } B_k > 0, \\
&= \bar{B}(1-\omega_0)^{-1} \\
\bar{F}_b &= \mu_2 \bar{B}_b (\mu_2 - \lambda_2)^{-1} = \bar{B}_b (1-\rho_2)^{-1} \\
\beta &= \text{average blocking rate over all } B_k > 0 \\
\tilde{\omega}_0 &= \text{task blocking probability } = 1 - \tilde{\rho}_1 (1-\rho_2)^{-1} \\
x_0 &= \tilde{x}_0 = -\ln(1-\tilde{\omega}_0)(\mu_2 - \lambda_2)^{-1} \\
\tilde{\mu} &= \mu_2 (1-\rho_1)
\end{aligned}
\tag{6.11}
$$

For model selection, minimally a value for ω_0 (or equivalent) must first be observed. A value for $\bar{B}$ also is often required.

6.4 Short Hyperperiods

As one might expect, when $\bar{B} \approx C$, the foreground aperiodic response times don't differ much from the background aperiodic response times. When $\bar{B} \approx C$, then $\tilde{\rho}_1 = \bar{B}H^{-1} \approx CH^{-1} = \rho_1$ and very similar criteria used in Chapter 5 can be applied here. Conjecture 6.4.1 defines conditions when the DSM can be used to describe the response time distribution of the aperiodic tasks.

Conjecture 6.4.1 (FGA SHM Conditions) *For foreground aperiodic scheduling, when the expected number of aperiodic arrivals during an average (DSM) response time is greater than or equal to the expected number of aperiodics discharged in a hyperperiod, the DSM provides reasonable estimates.*

More concisely, the DSM is reasonable when

$$\lambda_2 \bar{R}_{\text{dsm}} = \frac{\lambda_2}{(\mu_2(1-\rho_1)-\lambda_2)} \geq (H - \bar{B})\mu_2$$

or equivalently, when

$$\lambda_2 \geq \frac{\mu_2(1-\tilde{\rho}_1)(1-\rho_1)}{(H\mu_2)^{-1}+(1-\tilde{\rho}_1)} = \frac{\mu_2(1-\rho_1)}{1+(H\mu_2(1-\tilde{\rho}_1))^{-1}}. \tag{6.12}$$

A slightly more conservative but data independent test assumes $\tilde{\rho}_1 = 0$ and is given in Equation 6.13.

$$\lambda_2^* \geq \frac{\mu_2(1-\rho_1)}{1+(H\mu_2)^{-1}}. \tag{6.13}$$

As expected, the DSM applies to the foreground aperiodic model for only a subset of the system parameter ranges for which the DSM applies to the background aperiodic model. The data sampled suggests the use of λ_2^* in Equation 6.13 is reasonable if not preferable. Table 6.2 contains sample data with observations for several different system configurations. In the quantile column is the approximate quantile where the predicted and observed curves noticably diverge. To that point, the fit varies among samples.

H	C	λ_2	B	$\tilde{\rho}_1$	$\geq \lambda_m$	$\geq \lambda_m^*$	DSM fit	quantile
4	1	0.70	0.85	0.212	.569	.600	very good	x(.994)
8	2		1.59	0.199	.649	.667	moderate	x(.95)
32	8		4.70	0.147	.723	.727	moderate	x(.95)
64	16		6.66	0.104	.737	.738	poor	
128	32		8.53	0.067	.744	.744	very poor	
8	2	0.74	1.92	0.240	.644	.667	very good	x(.995)
16	4		3.78	0.236	.693	.706	good	x(.95)
32	8		7.14	0.223	.721	.727	moderate	x(.95)
64	16		13.09	0.205	.736	.738	poor	
128	32		24.39	0.191	.743	.744	poor	x(.80)
4	3	0.20	2.03	.508	.166	.200	very good	x(.99)
8	6		3.57	.446	.204	.222	moderate	x(1.00)
16	12		5.48	.342	.238	.235	poor	
32	24		10.94	.276	.240	.242	very poor	
8	6	0.24	5.43	.679	.180	.222	good	x(.99)
16	12		10.30	.644	.213	.235	good	x(.95)
32	24		18.94	.592	.232	.242	moderate	x(.95)
64	48		32.00	.500	.242	.246	poor	
128	96		48.90	.382	.247	.248	very poor	

Table 6.2: FGA SHM Selection Criterion Evaluation

6.4.1 Response Time Distribution

The arguments for Conjecture 6.4.1 are essentially the same as in Section 5.2.1. Response times means and variances can also be calculated using the techniques of observation 5.2.2. Figure 6.2 shows foreground aperiodic response time data when compared to the DSM. Figure 6.3 shows the corresponding Q-Q plots.

6.4.2 Aperiodic System Size Distribution

When hyperperiods are short the system size arguments in Section 5.3.1 can be adapted in the natural ways. This applies to system size distributions, as well as moments. Figure 6.4 shows sample data for aperiodic system size distributions when scheduled in the foreground discipline. Estimates are overly optimistic for small values of queue length, and improve for longer values of queue length. Not surprisingly,

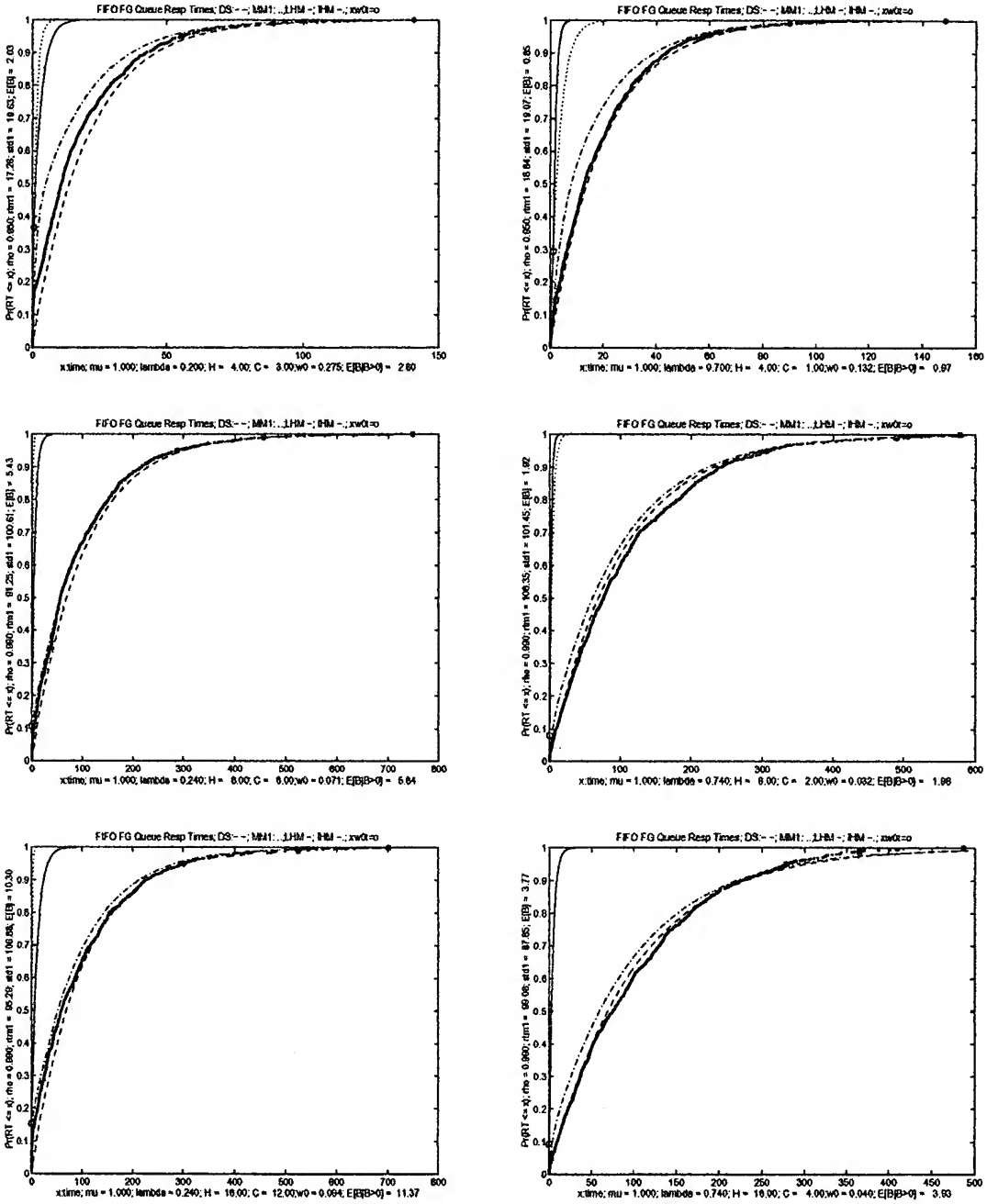


Figure 6.2: FGA SHM Response Time EDFs

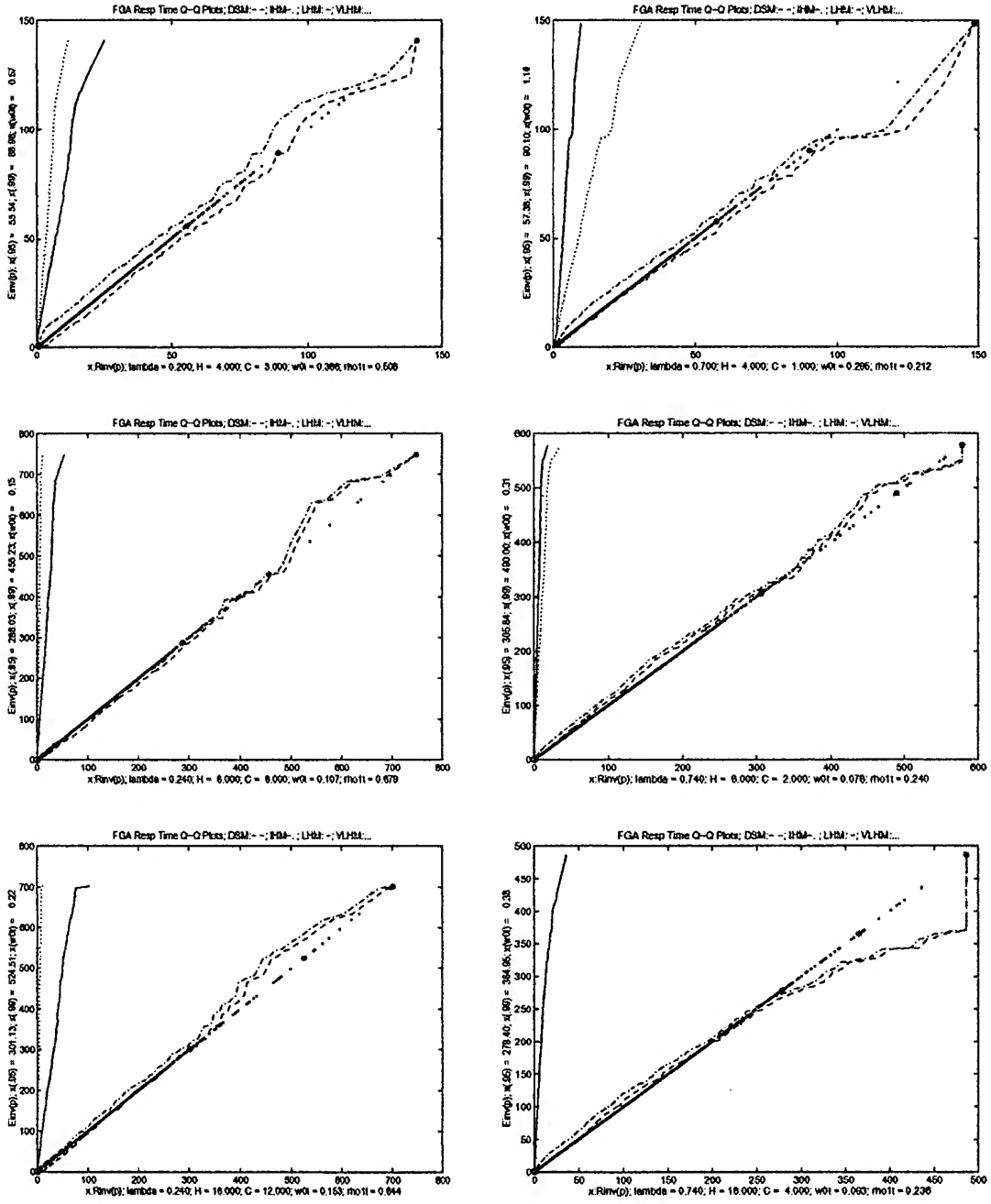


Figure 6.3: FGA SHM Response Time Q-Q Plots

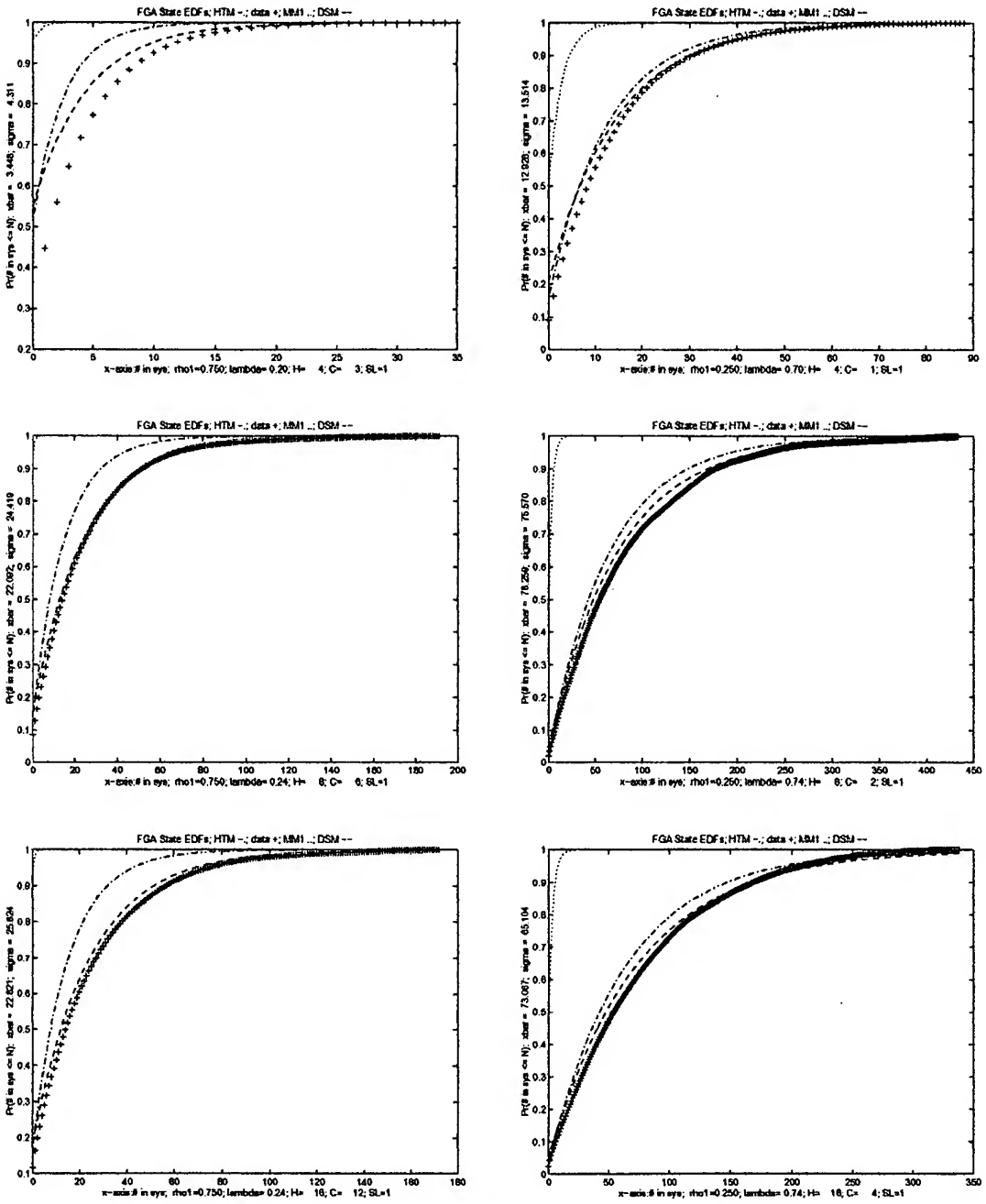


Figure 6.4: FGA SHM System Size EDFs

the overall fit for $\rho_1 = 0.25$ tends to be better than when $\rho_1 = 0.75$, and the overall fit improves as ρ increases.

6.5 Very Long Hyperperiods

When hyperperiods are very long, the aperiodic tasks essentially see the behavior of an M/M/1 queue. Based on empirical observations, we conjecture that when the effective periodic utilization is near zero, then the M/M/1 model is reasonable. This is formalized in Conjecture 6.5.1.

Conjecture 6.5.1 (FGA VLHM Conditions) *For foreground aperiodic scheduling, when the task blocking probability is less than 0.001, the M/M/1 response time model provides reasonable estimates.*

In symbols, the M/M/1 model applies when

$$\tilde{\rho}_1 = \frac{\bar{B}}{H} = \frac{(1-\omega_0)\bar{B}_b}{H} < \frac{0.001}{1-\rho_2}, \text{ or equivalently when } \tilde{\omega}_0 > 0.999.$$

Obviously,this criterion is somewhat arbitrary and can be specified to the precision measured in practice. Note when $\tilde{\omega}_0 \approx 1$, $\tilde{\rho}_1 \approx 0$ for all systems with $\rho_2 \not\approx 1$, which is consistent with intuition.

Table 6.3 contains some observed sample values under various parameter settings. In the Q-Q plots (some of which are included next) there tends to be uniformly good agreement to the value x_0, at which point there is often quite rapid divergence. Only when $\tilde{\rho}_1 \leq 0.0005$, are both the observed response time sample mean and standard deviation fairly close to the M/M/1 mean (m_r), and standard deviation (σ_r) which is seen in Table 6.4.

H	C	ρ	λ_2	ω_0	ω_p	B_b	$\tilde{\omega}_0$	$\tilde{\rho}_1$	MM-fit	quantile
4096	1024	0.99	0.74	0.516	0.484	123.63	0.944	0.012	poor	x(.95)
8192	2048			0.727	0.273	108.47	0.982	0.004	moderate	x(.95)
16384	4096			0.796	0.204	97.34	0.995	0.001	moderate	x(.995)
32768	8192			0.965	0.035	66.26	1.000	0.000	excellent	x(1.0)
512	128	0.95	0.70	0.785	0.215	22.46	0.991	0.009	poor	x(.80)
1024	256			0.875	0.125	26.49	0.995	0.003	good	x(.99)
2048	512			0.980	0.020	22.46	0.999	0.000	excellent	x(1.0)
2048	1536	0.99	0.24	0.608	0.392	126.10	0.968	0.024	poor	x(.97)
4096	3072			0.786	0.214	111.44	0.993	0.006	moderate	x(.993)
8192	6144			0.906	0.094	104.65	0.998	0.002	very good	x(.998)
16386	12288			0.959	0.041	149.38	1.000	0.000	excellent	x(1.0)
128	96	0.95	0.20	0.749	0.251	23.17	0.951	0.008	poor	x(.95)
512	384			0.947	0.053	29.21	0.996	0.003	moderate	x(.995)
1024	768			0.993	0.007	36.00	1.000	0.000	excellent	x(1.0)

Table 6.3: FGA VLHM Criterion Evaluation

H	C	ρ	λ_2	$\bar{m}_r$	$\hat{\sigma}_r$	$m_r = \sigma_r$
4096	1024	0.99	0.74	10.32	40.66	3.85
8192	2048			5.00	14.50	
16384	4096			4.16	9.45	
32768	8192			3.80	3.96	
512	128	0.95	0.70	4.22	7.44	3.33
1024	256			3.80	4.95	
2048	512			3.72	3.24	
2048	1536	0.99	0.24	4.21	25.45	1.32
4096	3072			1.99	9.96	
8192	6144			1.46	4.21	
16384	12288			1.31	1.31	
128	96	0.95	0.20	2.52	7.49	1.25
512	384			1.45	3.22	
1024	768			1.22	1.23	

Table 6.4: FGA VLHM Response Time Moments

6.5.1 Response Time Distribution

This is the idealized M/M/1 response time distribution where

$$R(x) = 1 - e^{-(\mu_2 - \lambda_2)x} \text{ for } x \geq 0.$$

Figure 6.5 shows some sample response time distributions with an M/M/1 response time overlay. Departures from the M/M/1 are often not readily detectable from the response time CDFs, but are from the Q-Q plots which are shown in Figure 6.6.

6.5.2 Aperiodic System Size Distributions

The idealized M/M/1 system size distribution where

$$\Pr(S \leq k) = 1 - \rho_2^{k+1} \text{ for } k \in \{0, 1, 2, ...\}$$

tends to be somewhat optimistic. Figure 6.7 shows several sample system size distributions with an M/M/1 system state overlay for the same simulation runs in Figure 6.5. Estimates are better for larger values of queue length.

6.6 Long Hyperperiods

The long hyperperiod model applies when some periodic blocking occurs, but blocking times are almost always limited to less than one periodic compute time. In other words, an aperiodic task's completion time contains one or fewer blocking periods, or equivalently when $\omega_m \approx 0$. When these conditions hold, the blocking time distribution is modeled as $\mathcal{E}(\beta)$, where β is one of the DSM or HTM rate parameters. We formalize conditions when the long hyperperiod model (LHM) is a reasonable approximation in Conjecture 6.6.1.

Conjecture 6.6.1 (FGA LHM Conditions) *For foreground aperiodic scheduling, when the probability of maximum blocking is suitably small and the VLHM does not apply, the LHM can be used to predict response time data.*

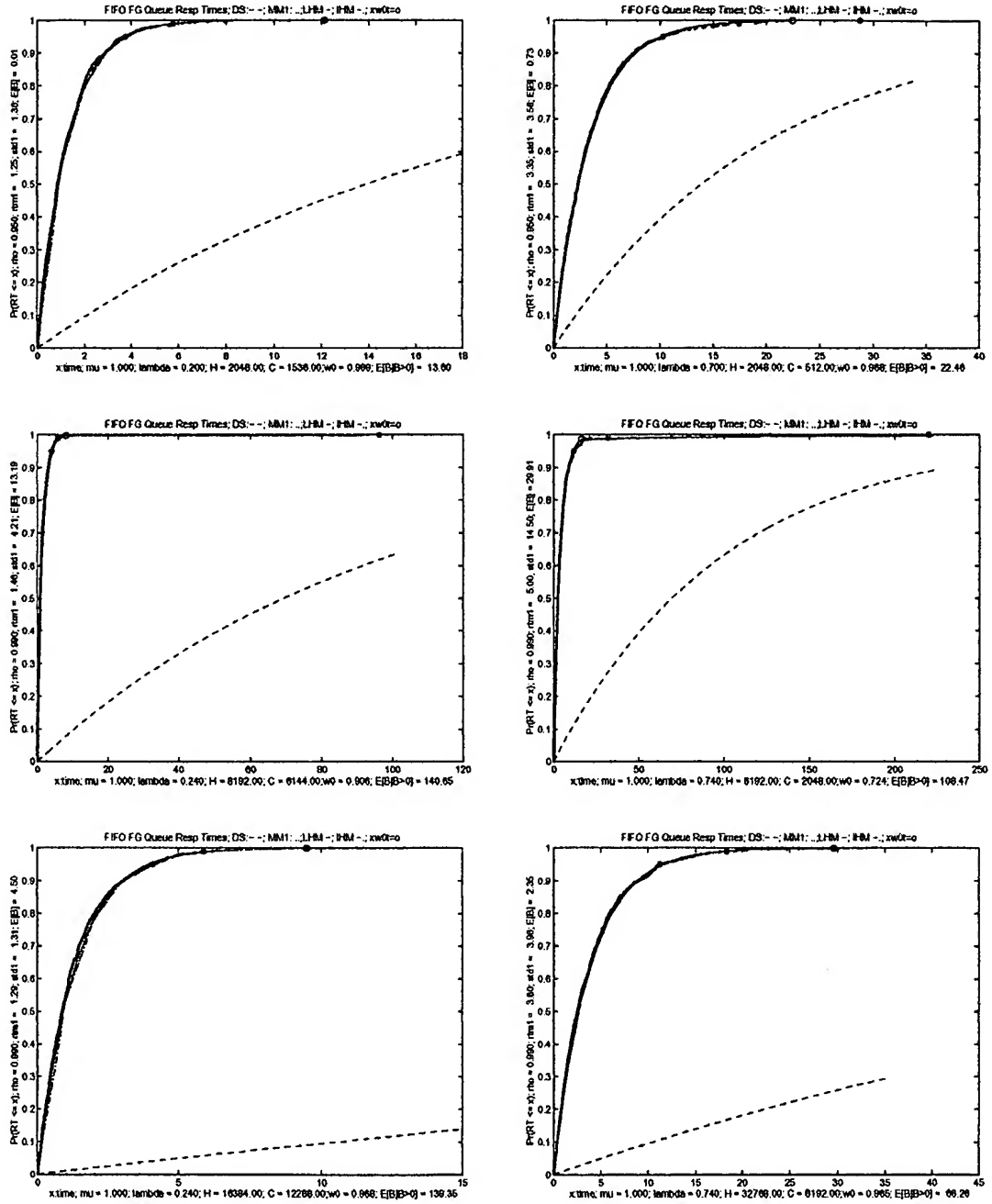


Figure 6.5: FGA VLHM Response Time EDFs

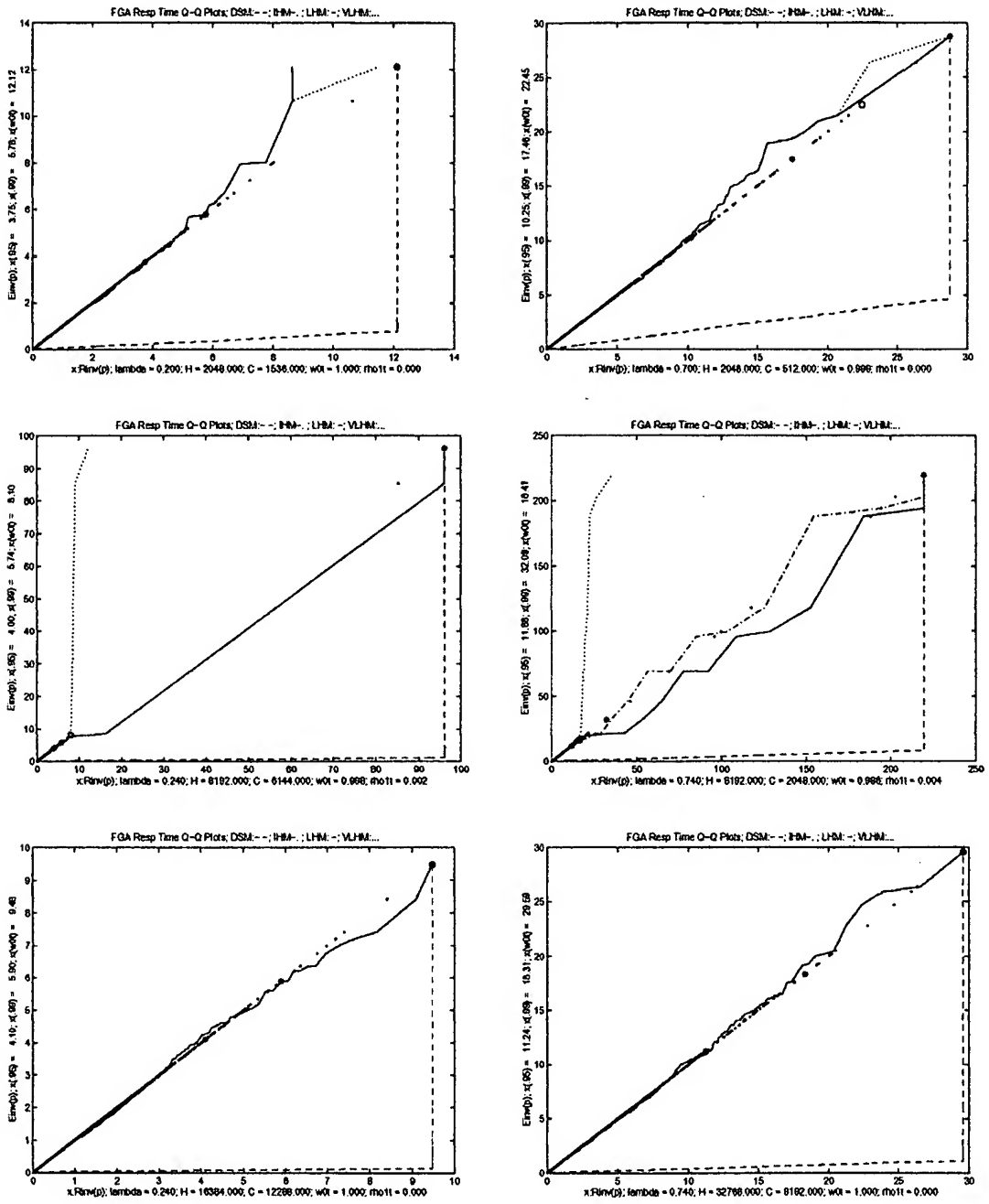


Figure 6.6: FGA VLHM Repsonse Time Q-Q Plots

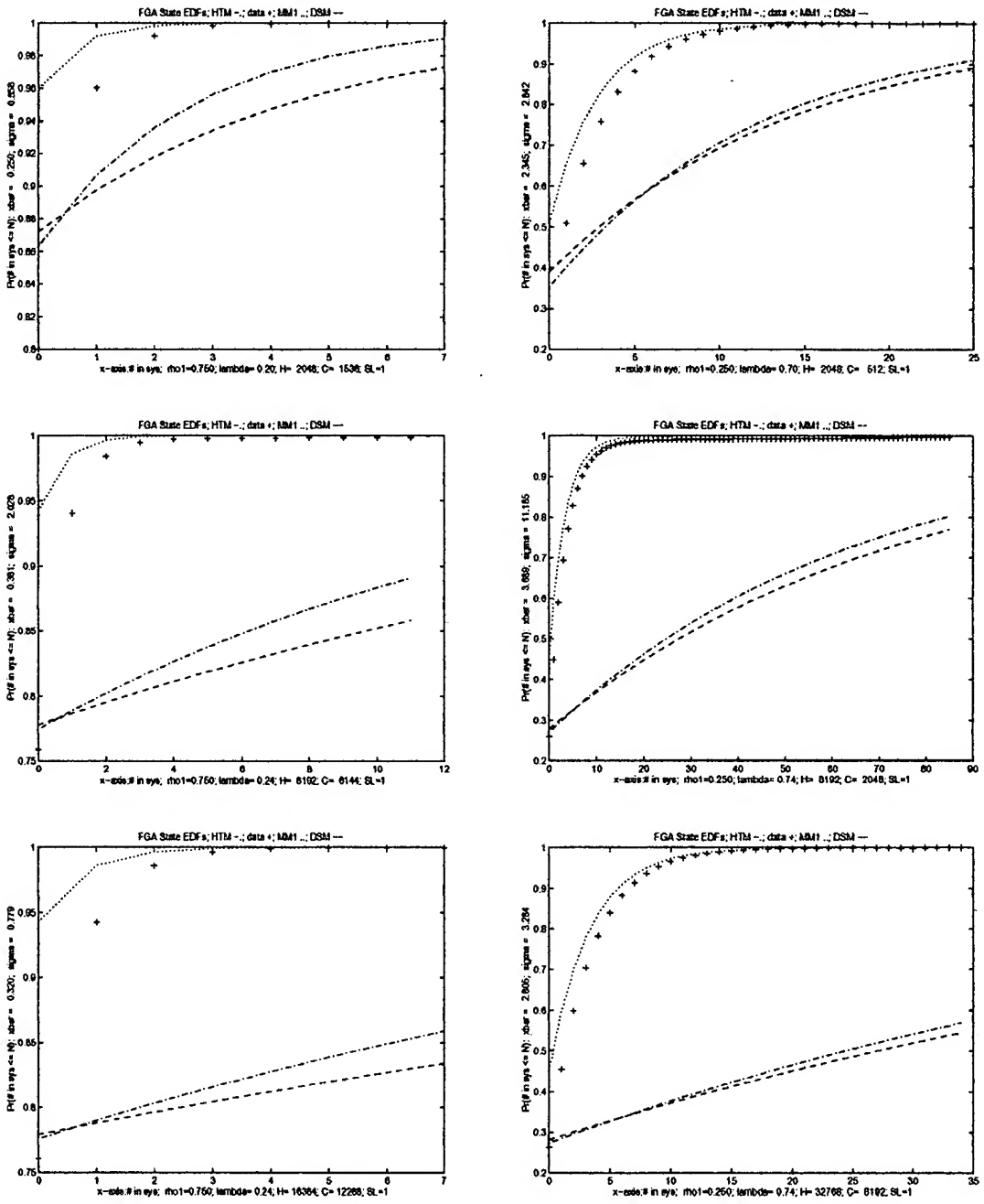


Figure 6.7: FGA VLHM System Size EDFs

More concisely, the LHM is reasonable when

$$\Pr(B = C) = \omega_m \approx 0 \text{ and } \tilde{\rho}_1 > \frac{0.001}{1-\rho_2}, \tag{6.14}$$

or equivalently, when

$$\Pr(B \leq C^-) = \omega_0 + \omega_p \approx 1 \text{ and } \tilde{\rho}_1 > \frac{0.001}{1-\rho_2}. \tag{6.15}$$

Empirical evidence suggests $\omega_m \leq 0.03$ *is suitably small. Under these conditions*[3]*, typically* $\bar{B}_b \approx \beta^{-1}$.

In the next several sections, we derive a model for describing aperiodic response time distributions under the long hyperperiod model conditions.

6.6.1 Aperiodic Queue Lengths during Blocking Times

When the periodic task is blocking aperiodic tasks for less than the full periodic compute time, the aperiodic tasks are in a state of heavy traffic. At the end of blocking intervals, the aperiodic queue length distribution can be approximated using the heavy traffic models of Section 3.4.1. Our data again shows the degraded server approximations are slightly less optimistic and also match the data slightly better. Both approximations are given below.

For the DSM, $E[N_d] = \lambda_2(\tilde{\mu}-\lambda_2)^{-1}$, and $\sigma_d = (\lambda_2\tilde{\mu})(\tilde{\mu}-\lambda_2)^{-1}$. The distribution of aperiodic system size at blocking interval ends, $N_{d,b}$, is given by Equation 4.2. For the HTM, let $N_{h,b}$ be the heavy traffic approximation of aperiodic queue length at

[3]When $\omega_0 \approx 1$, the number of hyperperiods observed can be small, in which case $\bar{B}_b < \beta^{-1}$ sometimes occurs.

λ_2	ρ_1	$E[N_d]$	σ_d	$\beta_{d,q}$	$\beta_{d,c}$	$E[N_h]$	$\beta_{h,q}$	$\beta_{h,c}$	LHM region
0.24	0.75	24.00	24.50	0.0417	0.01	15.125	0.066	0.016	$256 < H < 8192$
0.74	0.25	74.00	74.50	0.0135	0.01	65.125	0.015	0.011	$1024 < H < 16386$
0.20	0.75	4.00	4.47	0.2500	0.05	2.625	0.381	0.076	$64 < H < 512$
0.70	0.25	14.00	14.49	0.0710	0.05	12.625	0.079	0.055	$256 < H < 1024$
0.10	0.75	0.67	1.05	1.50	0.15	0.542	1.846	0.185	$32 < H < 128$
0.60	0.25	4.00	4.47	0.25	0.15	3.875	0.258	0.155	$128 < H < 256$

Table 6.5: Predicted Blocking Rates when $\omega_m = 0$

blocking periodic departures. Then

$$E[N_h] = \frac{\mu_2(1-\rho_1)^2 + \lambda_2}{2(\mu_2(1-\rho_1) - \lambda_2)},$$

so from Equation 3.6 $N_{h,b} \sim \mathcal{E}(E[N_h]^{-1})$.

Figure 6.8 shows aperiodic queue length distributions when sampled at the departures of blocking periodic tasks.

6.6.2 Blocking Time Distribution

When maximum blocking is rare (i.e. $\omega_m \approx 0$), the queue length is expected to grow linearly with blocking time. Specifically, during blocking times $N_2 \approx \lambda_2 B$ and hence $\beta_c \approx \lambda_2 \beta_q$. Table 6.5 lists predicted blocking rates and mean blocking times for several system configurations for which $\omega_m \approx 0$, and hence for which $e^{-\beta_c C} \approx 0$.

Conditional on some blocking occurring, the blocking time distribution is estimated by $\mathcal{E}(\beta_c)$. In other words, $\Pr(B \leq b \mid \text{given some blocking occurs})$ is given by

$$\Pr(B < b | B > 0) = B_p(x) \approx 1 - e^{-\beta_c b} = 1 - e^{-\beta b}. \tag{6.16}$$

Table 6.6 lists several observed values of estimates of β_c under a range of parameter

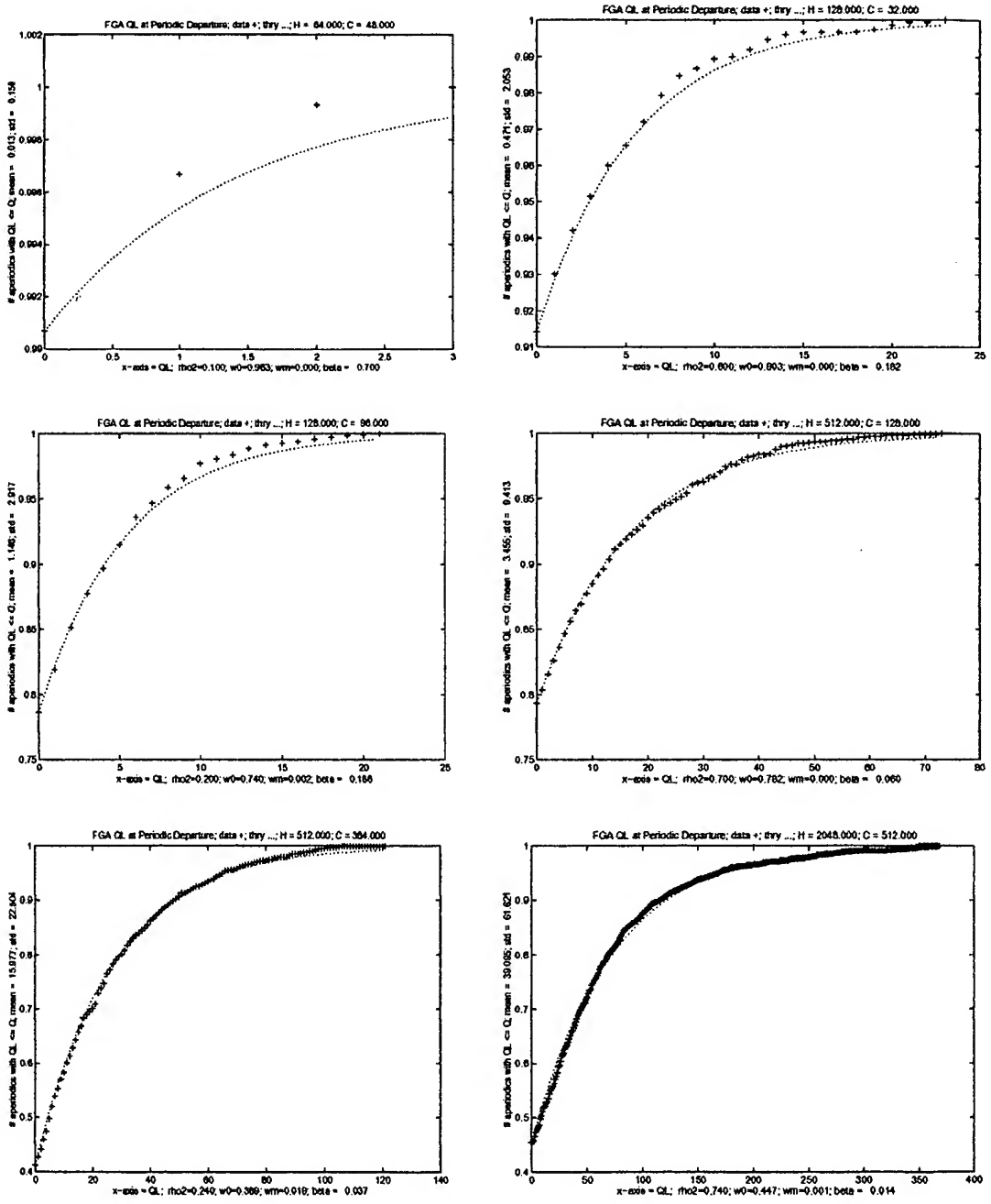


Figure 6.8: FGA LHM Queue Length EDFs at Blocking Periodic Departures

H	C	ρ	λ_2	B	$1-\omega_{0,c}$	β_c	ω_m	$\beta_{d,c}$
2048	512	0.99	0.74	59.772	0.554	0.009	0.002	.01
4096	1024			63.710	0.484	0.008	0.000	
8192	2048			38.325	0.273	0.007	0.000	
512	384	0.99	0.24	61.171	0.620	0.011	0.016	.01
2048	1536			40.736	0.361	0.010	0.000	
4096	3072			31.720	0.269	0.010	0.000	
8192	6144			9.465	0.098	0.012	0.000	
512	128	0.95	0.70	5.752	0.215	0.048	0.000	.05
1024	256			2.826	0.125	0.047	0.000	
128	96	0.95	0.20	5.722	0.259	0.049	0.000	.05
512	384			1.670	0.062	0.043	0.000	
128	32	0.85	0.60	0.751	0.098	0.130	0.002	.15
256	64			0.205	0.002	0.098	0.000	
512	128			0.010	0.001	0.096	0.000	
32	24	0.85	0.10	0.795	0.088	0.138	0.004	.15
64	48			0.269	0.022	0.102	0.000	
128	96			0.064	0.004	0.097	0.000	

Table 6.6: Observed Blocking Rates when $\omega_m \approx 0$

settings. Figure 6.9 shows some blocking time distributions for the foreground aperiodic scheduling discipline when $\omega_m \approx 0$. For the smaller utilizations (e.g. $\rho = 0.85$) and the longer hyperperiods, β_c tends to be smaller than estimated. Under these conditions, only a small number of blocking hyperperiods are observed and shorter blocking times are more likely to be observed than longer blocking times.[4]

6.6.3 Response Time Distribution

In this section, we develop a response time distribution approximation for systems satisfying the FGA LHM conditions. Define $x_0 = x_{\tilde{\omega}_0} = -\ln(1-\tilde{\omega}_0)(\mu_2 - \lambda_2)^{-1}$ so

[4]For $X_1, X_2, \ldots$ *iid* $\mathcal{E}(\beta)$, and $\bar{X}_k = (\sum_{j=1}^{k} X_j)/k$, then $P(\bar{X}_k \leq \beta^{-1}) = P(G_{\beta,k} \leq k\beta^{-1}) > 0.5$ for k small.

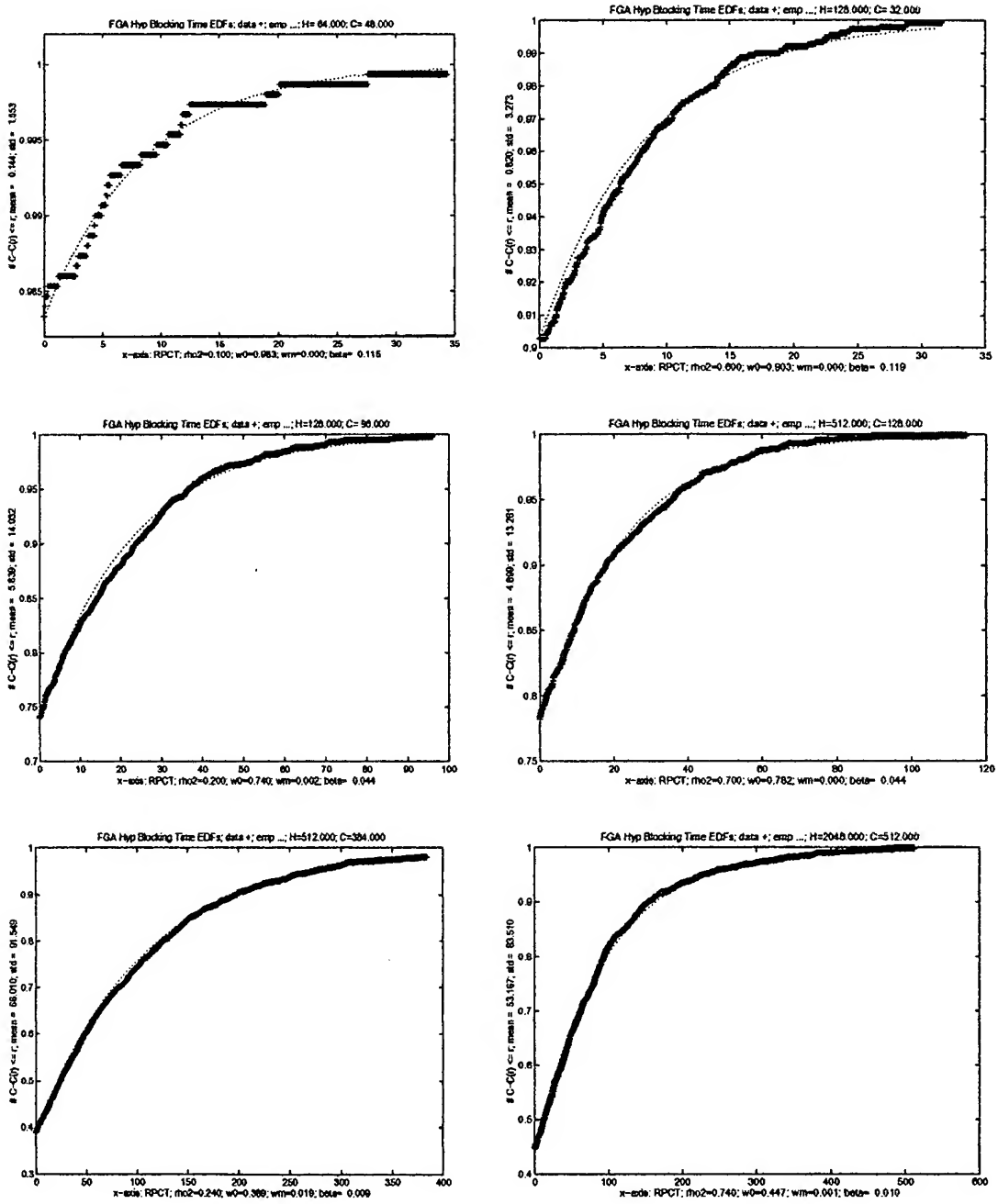


Figure 6.9: FGA LHM Blocking Time Distributions

that

$$R_m(x_0) = 1 - e^{-(\mu_2-\lambda_2)x_0} = \tilde{\omega}_0 = \omega_0 + (1-\omega_0)(1 - \frac{\beta^{-1}}{H(1-\rho_2)}) \approx 1 - \frac{\tilde{\rho}_1}{(1-\rho_2)},$$

where β is the mean blocking rate conditional on some blocking occurring and $\tilde{\omega}_0$ is the task blocking probability. Note that the long and very long hyperperiod response time models agree on $[0, x_0]$.

Recall from Section D.1.1 the development of the BGA LHM response time CDF when the blocking time distribution $B \sim \mathcal{D}(C)$. Define

$$R(x|B>0) = \int_0^C R(x|B=b)dB_p(b) = \int_0^C R(x|B=b)\beta_c e^{-\beta_c b}db,$$

where $R(x|B=b)$ is the BGA LHM response time distribution. For ease of exposition in this section, let α denote $[H(1-\rho_2)]^{-1}$. Then,

$$R(x|B=b>0) = \begin{cases} 0 & \text{for } x \le x_0 \\ 1-\alpha(b+x_0-x) & \text{for } x \in [x_0, b+x_0] \\ 1 & \text{for } x > b+x_0 \end{cases} \tag{6.17}$$

For $x \in [x_0, x_0+b]$, then $b \in [x-x_0, C]$. For $x > x_0+b$, then $0 \le b \le x-x_0$. Just as for the BGA LHM, we assume the support is non-zero only for $x \in [x_0, x_0+C]$. Thus,

$$R(x|B>0) = \int_0^{x-x_0} 1 \cdot dB_p(b) + \int_{(x-x_0)}^C (1-\alpha(x_0-x+b))dB_p(b),$$

where $B_p(b)$ is defined in Equation 6.16. Further computing gives

$$\begin{aligned} R(x|B>0) &= (1-e^{-\beta(x-x_0)}) + (1-\alpha(x_0-x))\int_{x-x_0}^{C} e^{-\beta b}db \\ &\quad -\alpha\beta\int_{x-x_0}^{C} be^{-\beta b}db \\ &= 1-\alpha\beta^{-1}e^{-\beta(x-x_0)} - (1-\alpha(\beta^{-1}+C+x_0-x))e^{-\beta C} \\ &\approx 1-\alpha\beta^{-1}e^{-\beta(x-x_0)} \text{ when } e^{-\beta C}\approx 0. \end{aligned} \tag{6.18}$$

Note that for suitably long C, $R(x_0|B>0) \approx 1-\bar{F}_b H^{-1}$, which is the probability an individual task (within a hyperperiod) will not block given some blocking occurs in that hyperperiod. For a task that experiences no blocking, the response time distribution is that of an M/M/1 queue for $r \leq x_0$. In a non-blocking hyperperiod, we make the simplifying assumption $\Pr(R > x_0|B=0) = 0$. For suitably long hyperperiods, this approximation can be justfied as follows. A bit of reflection reveals that $\Pr(R \leq x|B=0) \geq \Pr(R \leq x)$. So for $x > x_0$, $\tilde{\omega}_0 \leq \Pr(R \leq x) \leq \Pr(R \leq x|B=0)$. Once observing $\omega_0 \leq \tilde{\omega}_0$ an application of Conjecture 6.2.1 gives $\omega_0 \leq \tilde{\omega}_0 \uparrow 1$ as $H \uparrow \infty$.

Assuming all this, for $x \in [x_0, C+x_0]$, R_l is approximated by the mixture $R_l(x) = \omega_0 \cdot 1 + (1-\omega_0)(1-\alpha\beta^{-1}e^{-\beta(x-x_0)})$. By assumption, $e^{-\beta C} \approx 0$, so $R_l(x) = 1$ for $x \geq C+x_0$. This approximation is summarized in Equation 6.19. Once observing that $e^{-\beta(x-x_0)}$ is the probability that a new arrival sees a blocking time in excess of $x-x_0$, the similarity in form of Equation 6.19 with Equation 5.9 is noteworthy.

$$R_l(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1-e^{-(\mu_2-\lambda_2)x} & \text{for } x \in [0, x_0] \\ 1-\tilde{\rho}_1[(1-\rho_2)]^{-1}e^{-\beta(x-x_0)} & \text{for } x \in [x_0, C+x_0] \\ 1 & \text{for } x \geq C+x_0 \end{cases} \tag{6.19}$$

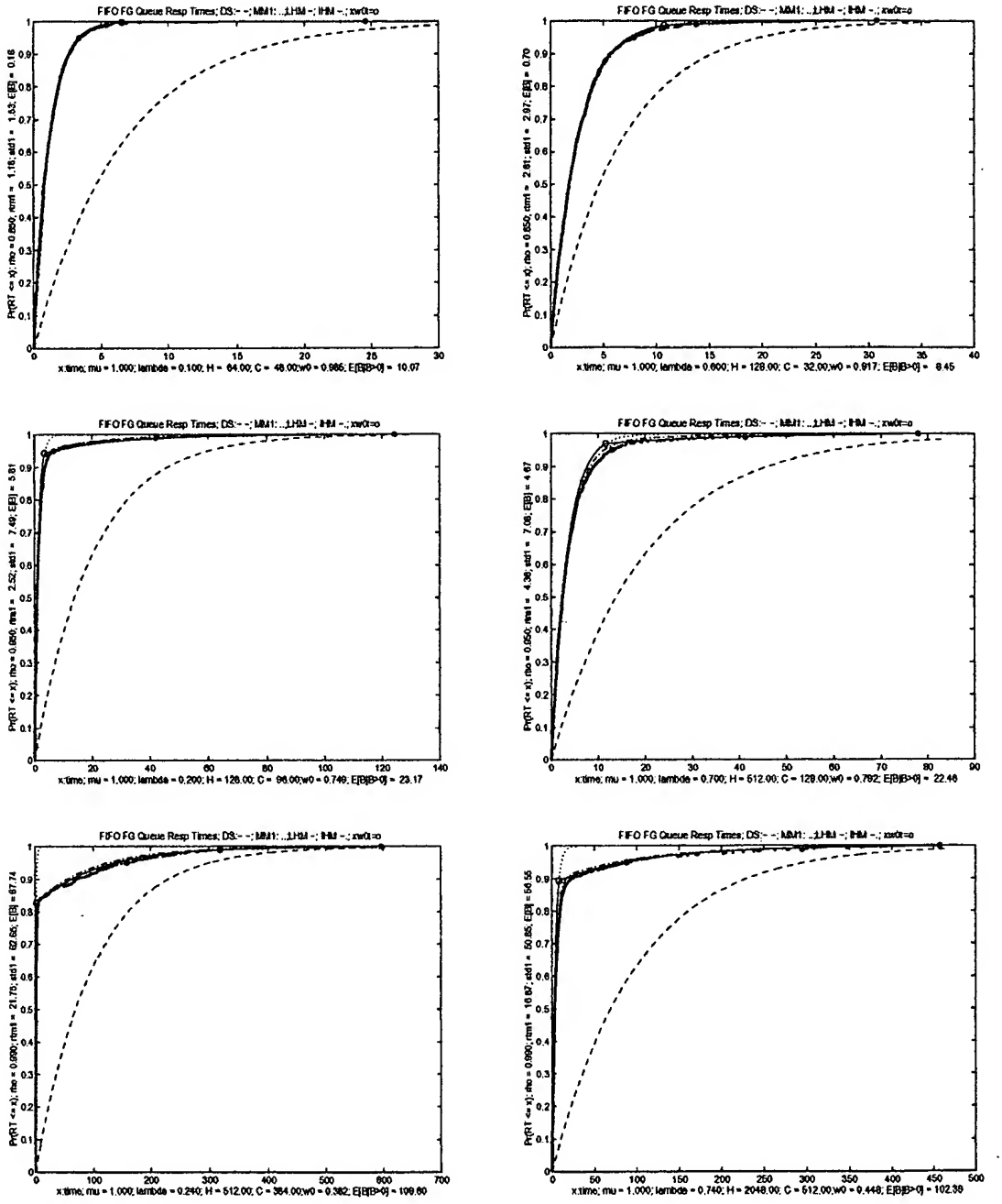


Figure 6.10: FGA LHM Response Time EDFs

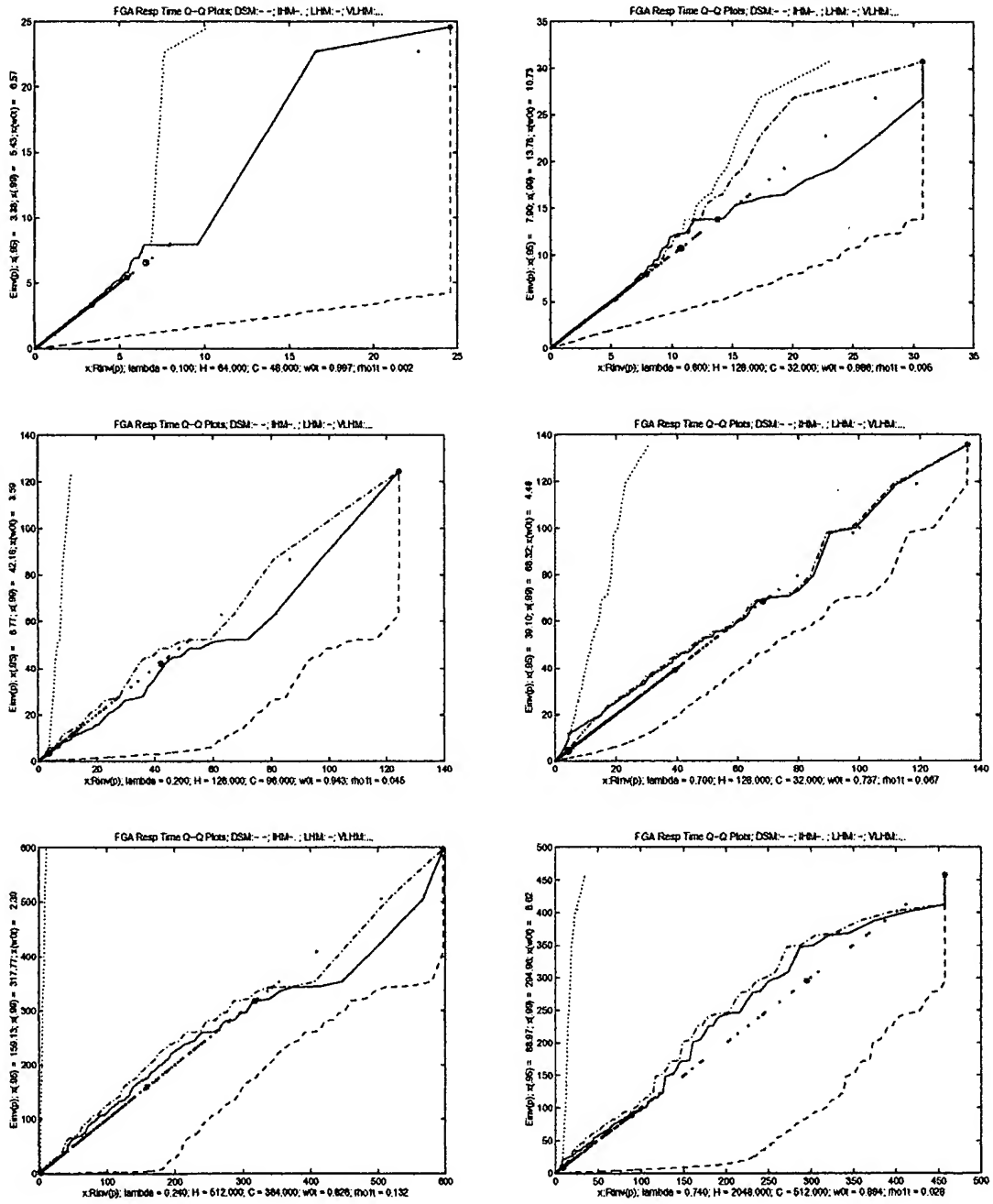


Figure 6.11: FGA LHM Response Time Q-Q Plots

Figure 6.10 shows several sample response time distributions plotted against the long foreground aperiodic hyperperiod response time model. Figure 6.11 shows the corresponding response time Q-Q plots.

6.7 Intermediate Hyperperiods

System configurations that fall in the intermediate hyperperiod category have each of ω_0, ω_p and $\omega_m \not\approx 0$. The number of response time bands will be two or greater. With the exception of the lowest band where some M/M/1 behavior can be exhibited throughout the hyperperiod, the response time curves fall strictly within the bands just as they did for background aperiodics.

The intermediate hyperperiod model (IHM) we are about to propose applies over a broad range of conditions, and when near the long hyperperiod model conditions (i.e. $\omega_m \approx 0$) is comparable to and slightly more optimistic (in the right tails) than the long hyperperiod model. The intuition behind the development of the FGA IHM response time distribution is that there is some M/M/1 behavior, but otherwise the response times are governed by the DSM model. When the FGA IHM and FGA LHM distributions are close, we choose the LHM since it is more conservative.

Conjecture 6.7.1 (FGA IHM Conditions) *In the foreground aperiodic scheduling model, the IHM is a candidate estimator of response times when λ_2 satisfies Equation 6.20.*

$$\lambda_2 \leq \frac{\mu_2(1-\rho_1)}{(H\mu_2)^{-1}+1} \text{ and } \omega_m > 0.03. \tag{6.20}$$

The left hand condition is simply the negation of the SHM conditions in Conjecture 6.4.1 and similarly the right hand condition is the negation of the LHM conditions given in Conjecture 6.6.1.

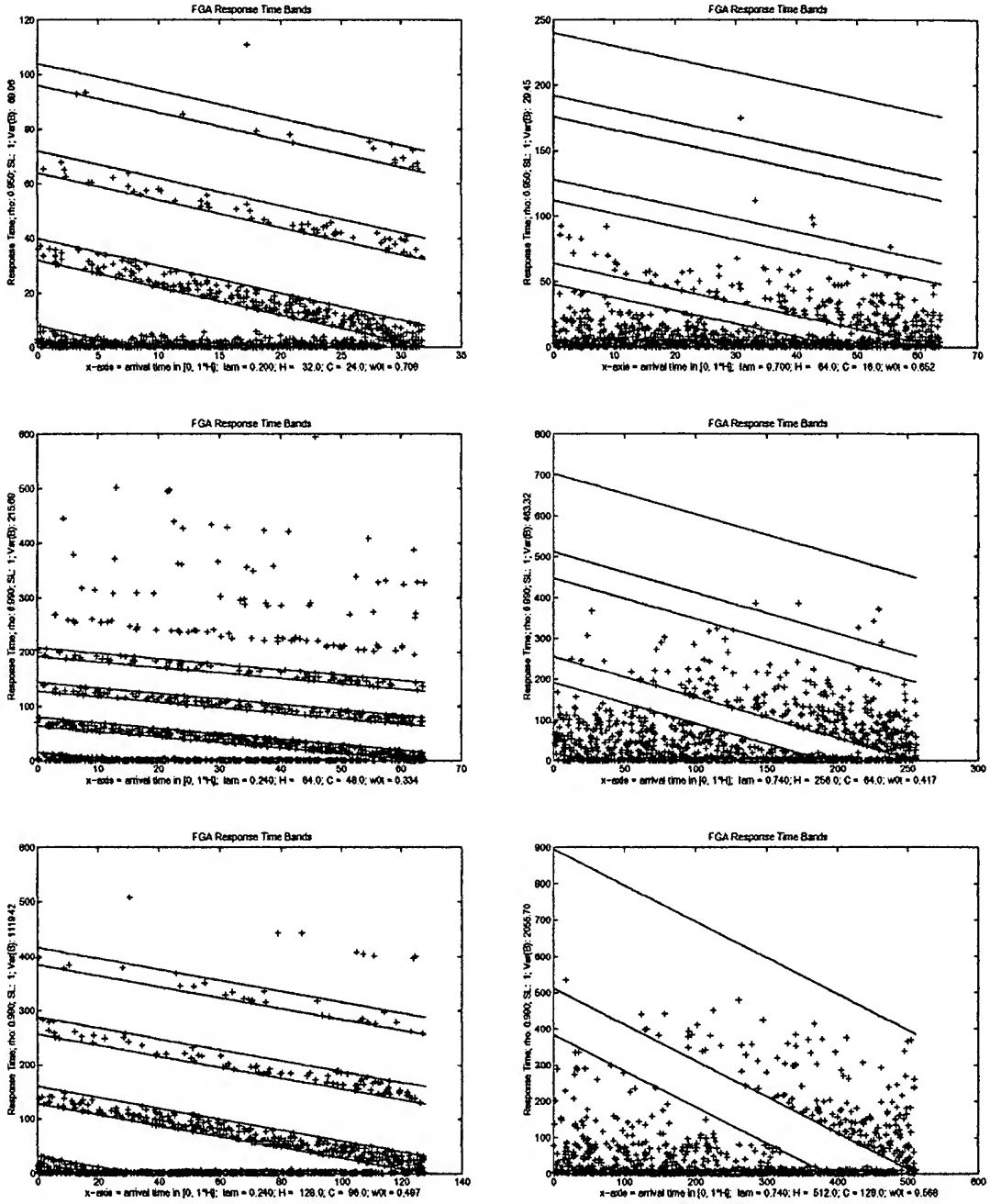


Figure 6.12: FGA IHM Response Time Data Bands

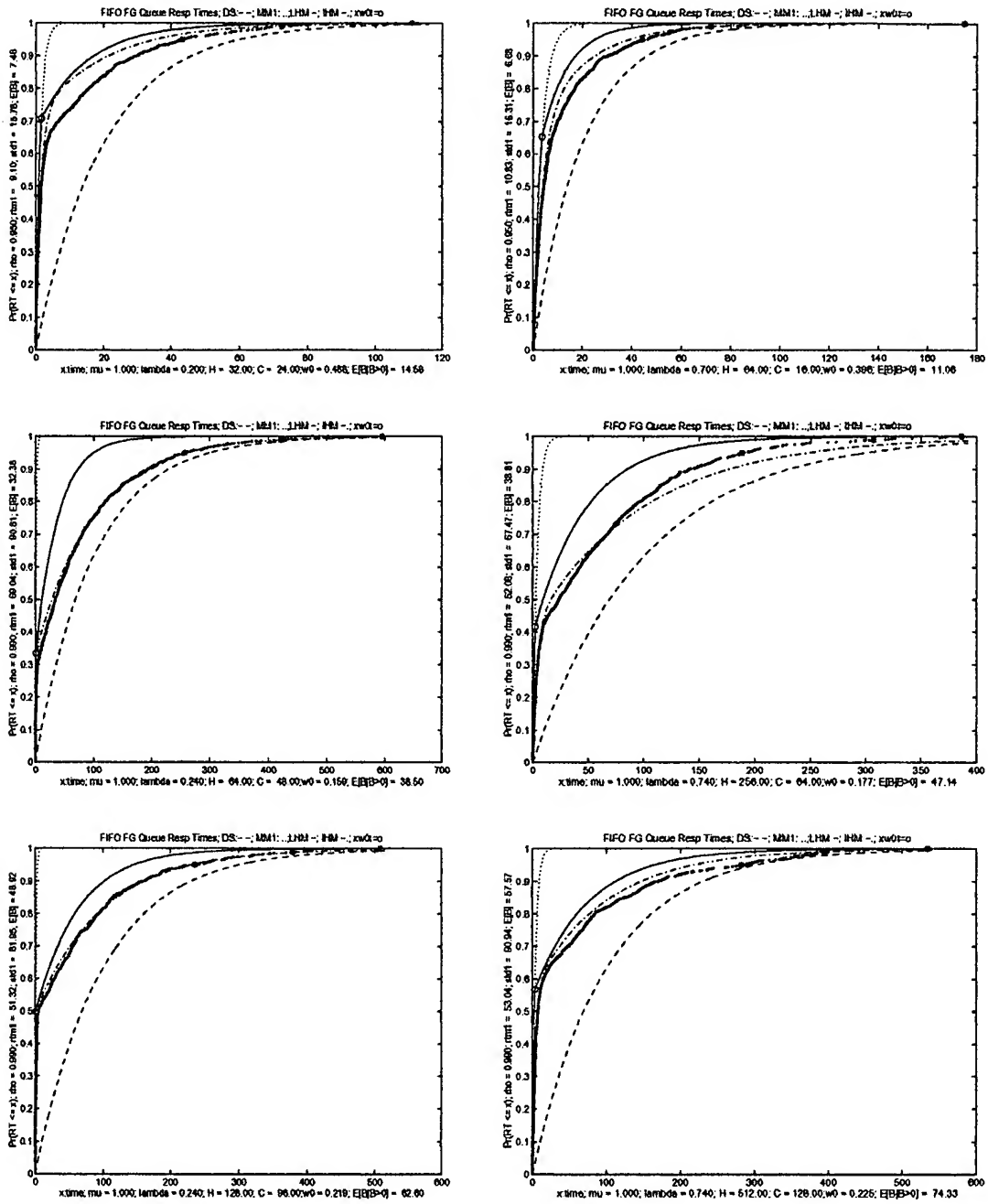


Figure 6.13: FGA IHM Response Time EDFs

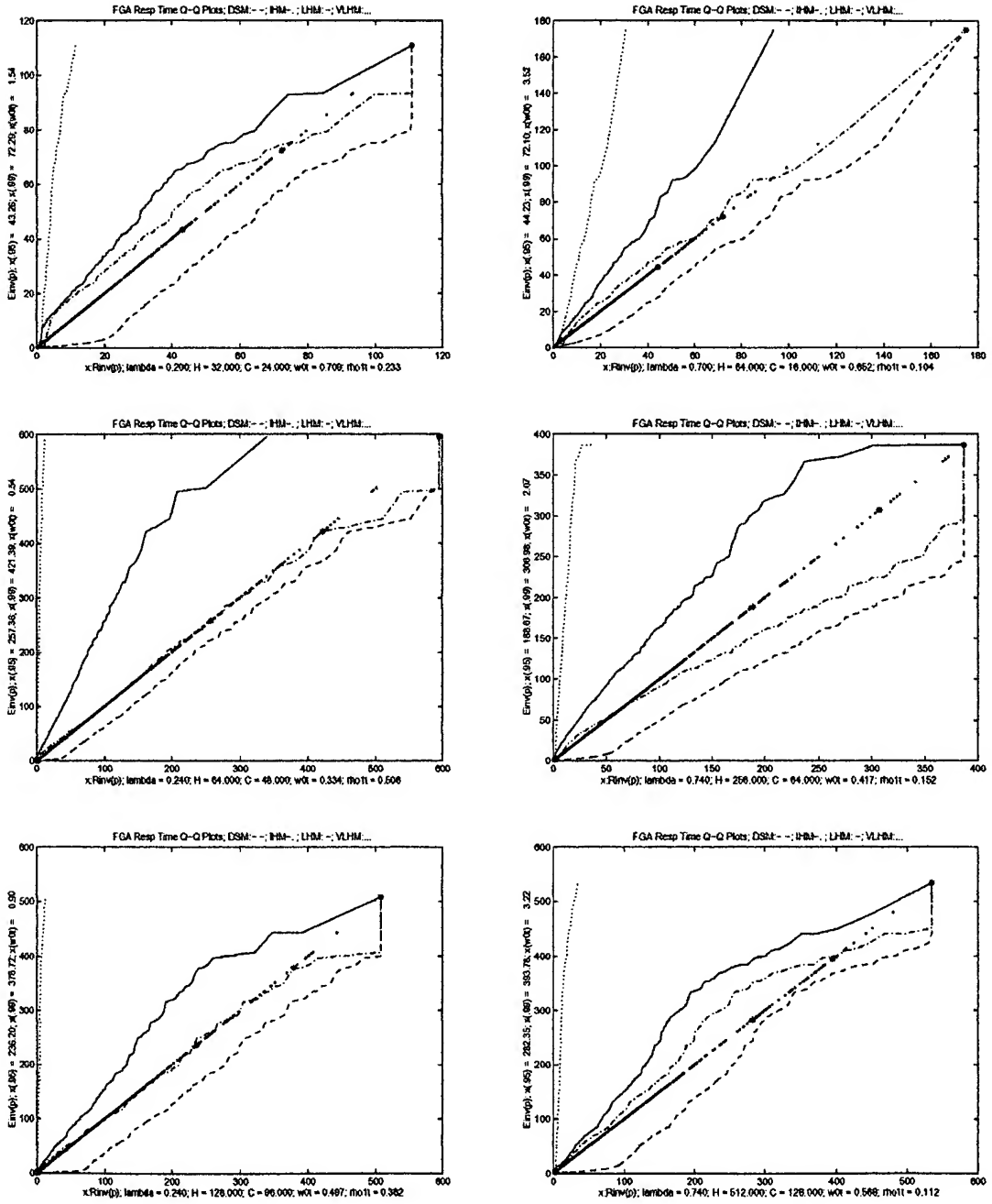


Figure 6.14: FGA IHM Response Time Q-Q Plots

Figure 6.12 shows some sample response time bands, with response times as the y-axis, and time of arrival within the hyperperiod as the x-axis. Equation 6.21 is an approximation for $R_i(x)$, the probability that the response time does not exceed x in an arbitrary hyperperiod.

$$R_i(x) = 1 - \tilde{\omega}_0 e^{-(\mu_2 - \lambda_2)x} - \tilde{\omega}_b e^{-(\tilde{\mu} - \lambda_2)x}. \tag{6.21}$$

The derivation is easy. For the task that experiences no sources of blocking, the response time distribution is that of an M/M/1. For the task that experiences blocking, we approximate the response time distribution as a DSM. This leads us to a mixture for a response time distribution,

$$\begin{aligned} R_i(x) &= \tilde{\omega}_0 R_m(x) + \tilde{\omega}_b R_d(x) \\ &= \tilde{\omega}_0 (1 - e^{-(\mu_2 - \lambda_2)x}) + \tilde{\omega}_b (1 - e^{-(\tilde{\mu} - \lambda_2)x}), \end{aligned} \tag{6.22}$$

where R_m is the M/M/1 response time distribution and R_d is the DSM response time distribution, which can be rewritten as Equation 6.21.

Figures 6.13 and 6.14 contain sample EDFs and Q-Q plots, respectively. The FGA IHM does not fit as well as the BGA IHM, but it also uses much less of the data (or equivalent) in its formulation. For $\rho_1 = 0.75$, the BGA IHM appears to fit moderately well. When $\rho \geq 0.95$, the IHM is often optimistic in the midrange quantiles when $\rho_1 = 0.25$. Equation 6.21 appears to work better for heavy traffic (e.g. $\rho = 0.99$) and tends to be optimistic for lighter traffic.

Chapter 7

Future Work

This thesis has looked at a few very special cases of the larger problem of *predictably* scheduling multiple sources of traffic, each with different service requirements, possibly including hard deadlines (which was the case for our periodic tasks). The general problem is not limited to a single server, but includes analyses for networks of queues.

Even for the special cases considered, we found we needed a collection of models to describe the response time (and in some cases system size) distributions of the aperiodic tasks. Within this collection, several models' and/or their selection criterion parameters were estimated using simulation data and not analytically derived. We begin by reiterating the analytically unknown parameters and conclude with a list of system generalizations which are candidates for future work.

7.1 Unknown Parameters

For serveral cases, we reduced the problem of analyzing certain aperiodic performance measures in a hard real-time system from one of determining entire distributions to one of selecting a parametric model. In a few cases, the parameters were analytically derived, but there are several cases where observed values for model parameters and/or model selection criteria were used. Finding mathematical descriptions for the model parameters enumerated below remains a challenge.

1. All models or selection criteria for the FGA scheduling discipline make (either direct or indirect) use of the value of $\omega_0 = \Pr(\text{no blocking within a hyperperiod occurs})$, which must now be empirically observed.

For the M/M/1 (VLH) model, the criteria is $\tilde{\omega}_0 > 0.999$. For both the IH and LH models, $\tilde{\omega}_0 = 1 - \tilde{\rho}_1(1-\rho_2)^{-1}$ is a part of the response time equation. For the SHM, $\tilde{\rho}_1$ can be used in the evaluation criterion, where $\tilde{\rho}_1 = \bar{B}H^{-1} = (1-\omega_0)\bar{B}_b H^{-1}$.

2. For the FGA scheduling discipline, the selection criterion boundary when choosing between the IHM and the LHM makes use of the value of $\omega_m = \Pr(\text{maximum blocking within a hyperperiod occurs})$, which must be observed. In some sense, blocking times of zero and C represent *boundary* conditions when the blocking time distribution is captured by a diffusion process. This suggests a solution technique for finding ω_0 may also be applicable for finding ω_m.

3. For the FGA scheduling discipline, the selection criterion boundary when choosing between the SHM and the LHM makes use of $\tilde{\rho}_1 = \bar{B}H^{-1}$. In this case $\omega_m \not\approx 0$, so $\bar{B}_b \not\approx \beta^{-1}$, hence $\bar{B}$ (or equivalently) must be observed.

4. For the BGA scheduling discipline, the IHM, the vector P^* had to be observed. We proposed obtaining these observed values via process/system simulation. This has the disadvantage that the values obtained need not be representative of a randomly chosen sample path. In particular, our values were exact since we used system simulation data to obtain them. Obtaining the values using process simulation potentially suffers from being sensative to initial state configurations and/or insufficient data for a good estimate (although not usually a problem in simulation).

   An alternative potentially viable approach might be to characterize the limiting distribution of the completion time process (which we have begun with our characterization of the blocking time distributions) and model the system as an M/G/1 queue. As we have seen, in principle we can obtain the LaPlace

transform for the response time distribution in terms of the completion time distribution. However, inversion of LaPlace transforms can require advanced techniques in complex analysis and/or numeric (inversion) methods. Once inverted, the integration to obtain the cdf might introduce additional numeric approximations. Also, any individual solution need not shed light on similar but different parameter settings.

In addition to requiring observations for model and/or model selection criteria parameters, sometimes models either did not fit well or no models were proposed.

1. For the FGA scheduling discipline, in the IHM range, the fit was not particularly good for "moderate" traffic. An approach similar to our background aperiodic IHM, using linear piecewise estimation for the response time bands would likely have produced better fitting models in non-heavy traffic. Since our focus was on heavily loaded systems and having done something very similar for the BGA IHM case, we opted to restrict focus on the applicability of models with closed form expressions.

2. In all cases, our investigations were restricted to $\rho_1 \in \{0.25, 0.75\}$. We suspect that similar criteria will hold for $\rho_1 \in [0.15, 0.85]$, and probably more broadly. These two values were chosen with the hope of capturing the impact of periodic task scheduling on aperiodic tasks when neither class largely dominates the system.

3. For the BGA scheduling discipline, system size models were not developed for the IHM case. For the FGA scheduling discipline, except for VLHM and SHMs, system size models were not developed.

The model boundaries defined by the model selection criteria will never be sharp, since in some sense the models are (topologically) close to one another near

the boundary conditions. However, in several cases our selection criteria were based on observations of parameter values with little more than an appeal to the intuitive reasonableness of their plausibility. With a more precise formulation (and greater understanding) of the model parameters, better motivated model selection conditions are likely to result.

7.2 System Model Generalizations

For the greatest chance of tractable solutions we have made the simplest of all assumptions about periodic and aperiodic properties for our system description. In all applications we have encountered, systems are considerably more complex than what we have assumed. In what follows, we sketch several complicating assumptions quite likely to occur in practical situations. We restrict focus to a single server, although the predictable behavior of networks is also of great interest in practice.

7.2.1 Multiple Periodic Streams

In hard real-time systems there are typically many periodic tasks which may have different periods. Perhaps the best known preemptive fixed-priority scheduling algorithm for periodic tasks in hard real-time systems is Rate Monotonic Scheduling (RMS,[27]). In RMS, there is a set of n periodic tasks with periods $T = (T_1, T_2, ..., T_n)$ and compute times $C_1 = (C_{1,1}, C_{1,2}, ..., C_{1,n})$. The execution of the periodic tasks cycle every hyperperiod (H), where H is the least common multiple of the periods, or $H = \text{lcm}\{T_1, T_2, ..., T_n\}$. The periodic utilization is defined by, $\rho_1 = \sum_{j=1}^{n} C_{1,j} T_j^{-1}$.

When scheduling aperiodic tasks in RMS systems, a background aperiodic policy was often adopted. Within the last decade, several algorithms have been

devised to give improved response time to non-critical aperiodic tasks while guaranteeing the hard deadline requirements of the periodic tasks. In our opinion, the *slack stealer* ([24]) provides the most robust design basis for algorithms in this class. To our knowledge, mathematical descriptions of aperiodic task behavior when scheduled using these algorithms remains largely undeveloped. When there is a single periodic task, our foreground aperiodic scheduling discipline is a special case of the slack stealer (and also of an algorithm known as the sporadic server).

We now consider the effects of multiple periodic tasks on aperiodic response times when compared to a single periodic task for fixed H and ρ_1 (we are considering more than just RMS). As an illustration, consider an example with which we are already familiar. Figure 7.1 shows cumulative aperiodic timeline availability as a function of time for two different task sets. The solid line corresponds to a single periodic task with period H_1 and compute time C_1. The dot-dash line also corresponds to the case a single task with period $H_2 = H_1/m$ and compute time $C_2 = C_1/m$, for m a positive integer. An alternative view of the dot-dash line is there are m tasks evenly spaced tasks throughout a hyperperiod of length H_1 each with compute time $C_2 = C_1/m$. The bottom lines are the exact aperiodic timeline availabilty that would be seen by an aperiodic stream when using the BGA scheduling discipline. The top lines are the maximums of the aperiodic timeline availability when using the FGA scheduling discipline. The maximums occur only when the system is constantly busy.

Let S_1 be a schedule with a single periodic task, and let S_2 be a schedule with multiple periodic tasks but with equal H and ρ_1. It is an easy exercise to show that for any aperiodic arrival process, when using a foreground aperiodic scheduling discipline, aperiodic response times in S_1 will be smaller than aperiodic response times in S_2 for the simple reason that the maximum aperiodic timeline in S_1 is no less than the maximum aperiodic timeline in S_2. Note that in the absence of aperiodic arrivals, the aperiodic timeline will decrease, and S_1 also has the ability to decrease more than

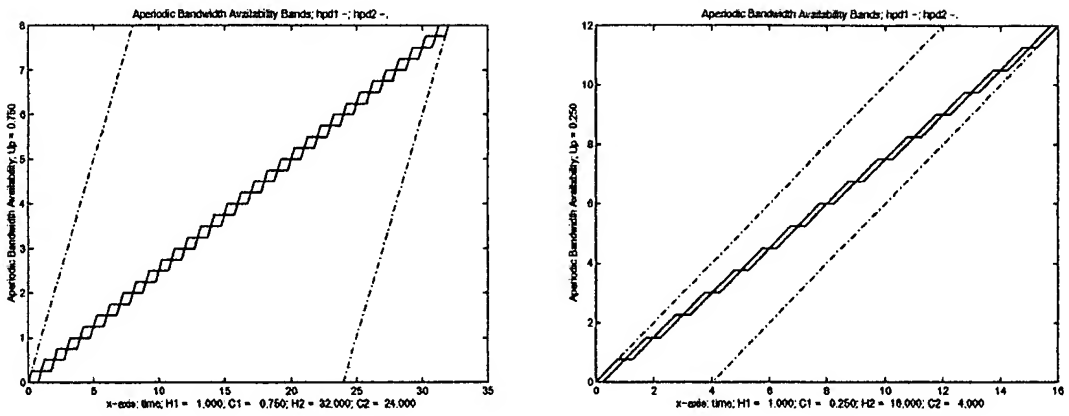


Figure 7.1: Aperiodic Timeline Availability

S_2 implying that the available slack in S_1 is always greater than that in S_2. Hence, our response time results for FGA scheduling are optimal compared to all other fifo aperiodic disciplines with equal H, ρ_1 and guaranteed hard deadlines for periodic tasks.

One might initially think a similar argument applies to S_1 and S_2 when using the background aperiodic scheduling discipline. However this need not be the case since both the time of aperiodic task arrivals and the amount of pending work determine response times. It is easy to construct sample paths for which the response times in S_1 are better than those in S_2. For example, consider an amount of aperiodic work W_a, $H_2 - C_2 < W_a < H_1 - C_1$ arriving exactly at time C_1. In S_1, no blocking occurs and in S_2 some blocking occurs. One must average over all sample paths and look at the limiting distributions to decide if S_1 performs better than S_2 in terms of providing a stochastically smaller response time distribution to aperiodics. Despite this counterexample, we suspect that the BGA LHM is pessimistic compared to other models with equal hyperperiods and equal periodic utilization when averaging over all sample aperiodic arrival paths (perhaps assuming that in the long run, aperiodics will arrive roughly equally throughout the entire hyperperiod).

When even spacing is applied in S_2 as illustrated in Figure 7.1, the LHM is transformed to a DSM as $m \uparrow$. Suppose $\mu_2 = 1$ and consider the BGA SHM criterion rewritten as

$$H < \frac{\lambda_2}{(1-\rho_1)(1-\lambda_2)}.$$

Alternatively, rewrite the BGA LHM criterion as

$$H > \frac{k_{\rho_1}(1-\lambda_2)}{(1-\lambda_2-\rho_1)^2}.$$

Assuming the validity of these two criteria, this tells us that if under configuration S_1 the LHM holds, then we can find an integer m such that S_2 becomes the DSM analog of S_1. Recall that when the BGA LHM holds, the DSM was much too optimistic.

Looking at non-uniform distributions of periodic execution timelines remains for future work for both foreground and background aperiodic scheduling disciplines. Another practical area for investigation is allowing the periodic exectuion time to be random but bounded (by C). Unused periodic compute time can be *reclaimed* and reallocated to aperiodic tasks. However, an analysis using the worst case periodic compute times will provide a stochastically larger aperiodic response time distribution for a single server.

7.2.2 Different Aperiodic Interarrival/Service Distributions

We have assumed that both the aperiodic interarrival and service time distributions are exponential. If all our arguments were based on heavy traffic theory (rather than the degraded server model for a fifo M/M/1 queue - which we often opted for because they were slightly more pessimistic in heavy traffic and much more accurate in moderate traffic), then we suspect similar arguments and results would apply under conditions of heavy traffic.

The only calculation used that made very explicit use of the the M/M/1 assumption was the mean return time to the M/M/1 interval in the BGA LHM. When $\rho \approx 1$, there are no changes in the BGA IHM, and the BGA DSM would instead use parameters derived using heavy traffic theory. Our analysis of the FGA scheduling discipline makes use of the known response time distribution for the M/M/1 queue when describing task response times that do not block. Since these tasks are not in heavy traffic, a heavy traffic approximation is not appropriate in this region. Any techniques applicable for estimating response time distributions in the absense of periodic processes are applicable for this region, which does not define the right tails of the response time distribution.

7.2.3 Multiple Aperiodic Streams

We considered only a single source of aperiodic traffic. In practice, there might be multiple sources of aperiodic traffic. If there are priorities among the aperiodic streams, the analysis would proceed first for the highest priority aperiodic stream, and then for the next highest priority aperiodic stream, etc. If all aperiodic streams have lower priority than a single periodic stream, then depending on total utilization, the highest priority aperiodic stream might be analyzed using techniques developed here when the criterion hold. If the system is near saturation, the lowest priority stream will be in a state of heavy traffic and might be analyzed using heavy traffic theory.

When there are two or more independent aperiodic streams, each with a Poission arrival process and fixed priorities, a completion time distribution for low(er) priority aperiodic traffic using techniques similar to those presented in [17] can be derived. To develop the response time analysis for the low(er) priority aperiodic stream in a system with a periodic traffic stream and higher priority (or priorities) aperiodic

traffic, the completion time distribution would be used to describe the low priority aperiodic traffic's service time distribution. When not all aperiodic arrival processes are Poisson and service distributions are general, a tractable exact analysis (in the absense of periodic traffic) does not exist in any general case. Nonetheless, determining sensitivities to the various assumptions would be useful when estimating response times for real applications.

7.2.4 Different Aperiodic Service Disciplines

By assuming a FIFO aperiodic task queue, aperiodic tasks are served in order of arrival. It is a more challenging problem when individual tasks are allowed to have distinct deadlines (drawn from a common distribution), in which case a more optimal scheduling discipline might assign (dynamic) priorities (within a class) giving highest priority to the task with the shortest time to deadline to minimize the number of *missed* deadlines.

The well known *earliest deadline first* (EDF) scheduling discipline assigns highest priority to the task with the smallest deadline and uses preemption to ensure that the task with the highest priority is receiving service. The optimality of EDF has long been known ([27]) to minimize the number of missed deadlines. Still, the response time distribution for tasks with individual deadlines remains unsolved except in very special cases.[1]

An alternative to EDF is the least laxity (LL) scheduling discipline which is non-preemptive (among aperiodic tasks). Each task has a laxity, which is the maximum time it can wait before beginning execution. The LL discipline will select the task with the smallest laxity only at aperiodic task departure times (assuming the server is busy; arrivals finding an idle server are started right away). Note that

[1] When all task deadlines are the same constant, then fifo is the same as edf.[25]

the LL as defined, does not make use of execution times in its decision rule for which task to schedule. Certain optimality properties about LL are also known ([30]).

In more recent work ([25],[26]), a closed form distribution for *leadtimes* when conditioning on a fixed queue length is derived for the EDF scheduling discipline under conditions of heavy traffic. The leadtime of a task is the deadline minus the current time. If the queue length distribution were known, then the unconditional leadtime distribution could be calculated which could be used to advise the developer of the probability that an individual task would miss its deadline. There are no deadline guarantees for select traffic in this model. Our (partial) queue length analyses and region selection criteria might be useful in determining when the conditional leadtime analysis results could be expected to apply to mixed hard deadline periodic and aperiodic sheduling problems, and perhaps to approximate unconditional leadtime distributions.

Appendix A

Notation

Table A.1 summarizes performance variables, their distributions and parameters used throughout this thesis. Much of this notation is first introduced (in the context of a single class of traffic) in Chapter 2.

Notation	Description
$\tau_{i,n}$	The n^{th} task of class i to arrive to the system. Periodic tasks have $i = 1$ and aperiodic tasks have $i = 2$.
H	The (hyper)period of the periodic tasks.
C	The compute time a periodic task.
$T_{i,n}$	A random variable defining the n^{th} interarrival time within the class i arrival stream. $\{T_{i,n}\}$ forms an *iid* sequence of random variables.
$A_i(t)$	The class i interarrival time distribution. $\Pr(T_{i,n} \leq t) = A_i(t)$.
λ_i	The class i interarrival rate. $E[T_{i,n}] = (\lambda_i)^{-1}$.
$\sigma^2_{a,i}$	The class i interarrival time variance. $\mathrm{Var}[T_{i,n}] = \sigma^2_{a,i}$.
$X_{i,n}$	A random variable defining n^{th} class i service time. $\{X_{i,n}\}$ forms an *iid* sequence of random variables.
$B_i(x)$	The class i service time distribution. $\Pr(X_{i,n} \leq x) = B_i(x)$.
μ_i	The class i service rate. $E[X_{i,n}] = (\mu_i)^{-1}$.
m_i	The mean service time for a class i task. $E[X_{i,n}] = m_i$.
$\sigma^2_{b,i}$	The class i service time variance. $\mathrm{Var}[X_{i,n}] = \sigma^2_{b,i}$.
ρ_i	The class i utilization. $\rho_i = \lambda_i \mu_i^{-1}$.
ρ	The system utilization. $\rho = \rho_1 + \rho_2$.

Table A.1: Performance and Limiting Distribution Variables

State variables are shown in Table A.2. It is common to use the *same mnemonic designations* for process state variables and for limiting distribution variables (since they will often converge to a random variable defining the steady state distribution). The context of their use should make their meaning clear. State variables first appear in Chapter 2 in the context of a single class of traffic. They are later defined

for multiple classes in Chapter 3. The bottom portion of Table A.2 defines some centered and scaled random functions used in weak convergence arguments. Random functions defining the n^{th} in a converging sequence of processes are superscripted.

Notation	Description
i	A class identifier index. Typically, for $i = 1$ the class is periodic, and for $i = 2$ the class is aperiodic.
$A_i(t)$	The number of class i arrivals in $[0,t]$. $A_i(t) = \max\{k \in \{0,1,2,...\} \mid \sum_{j=1}^{k} T_{i,j} \leq t\}$.
$N(t) = N_t$	The number of tasks in the system at time t. $N(t) = N_1(t) + N_2(t)$.
$B_i(t)$	$B_i(t) = \max\{k \in \{0,1,2,...\} \mid \sum_{j=1}^{k} X_{i,j} \leq t\}$, The number of class i departures in $[0,t]$ given full processing capacity.
$X_i(t)$	The cumulative class i workload requested in $[0,t]$. $X_i(t) = \sum_{j=1}^{A_i(t)} X_{i,j}$.
$I_i(t)$	The cumulative class i idle time in $(0,t)$ which is the amount of time available for class i tasks during which no class i customers were present.
$D_i(t)$	The number of class i departures in $[0,t]$. $D_1(t) = S_1(t - I_1(t))$ and $D_2(t) = S_2(I_1(t) - I_2(t))$.
$N_i(t)$	The number of class i tasks in the system at time t. $N_i(t) = A_i(t) - D_i(t)$, when $N_i(0) = 0$.
$Q_i(t)$	The number of class i tasks queued at time t. The relationship between $N_i(t)$ and $Q_i(t)$ varies with different priority schemes.
$Q(t) = Q_t$	The number of tasks queued at time t. $Q(t) = Q_1(t) + Q_2(t)$.
$W_i(t)$	The unfinished class i work at time t. With multiple classes, the interpretation of a virtual waiting time must be adapted.
$W(t) = W_t$	The unfinished work at time t. $W(t) = W_1(t) + W_2(t)$.
A_i^n	$A_i^n = n^{-\frac{1}{2}}[A_i(nt) - \lambda_i nt]$, the n^{th} scaled and centered class i arrival counting process.
X_i^n	$X_i^n = n^{-\frac{1}{2}}[X_i(nt) - \rho_i nt]$, the n^{th} scaled and centered class i cumulative workload request process.
S_i^n	$S_i^n = n^{-\frac{1}{2}}[\sum_{j=1}^{\lfloor nt \rfloor}(X_{i,j} - m_i)]$, the n^{th} scaled and centered class i service time process.
B_i^n	$B_i^n = n^{-\frac{1}{2}}[B_i(nt) - \mu_i nt]$, the n^{th} scaled and centered class i arrival counting process.

Table A.2: Notation: State Variable Descriptions

Appendix B

The M/M/1 Queue

Table B in this appendix lists many well known facts about the M/M/1 Queue, most of which have been used in this thesis. Most all of the listed results can be found in [35] (Takacs). Alternative references for many of these results are [1], [21] and [22]. A single theorem is also stated, since it is referenced from Appendix C.

Lemma B.0.1 (The mean and variance of a busy cycle.) *The mean of an idle period is λ^{-1}, and the mean of a busy period is $(\mu-\lambda)^{-1}$. So the mean of the busy cycle is*

$$E(\mathrm{BC}) = \frac{\mu}{\lambda(\mu-\lambda)}.$$

The variance of an idle period is λ^{-2} and the variance of a busy period can be shown to be $(1+\rho)[\mu^2(1-\rho)^3]^{-1}$ ([31]). Since the busy and idle cycles are independent (by the assumption of Poisson arrivals), the variance of the busy cycle (an adjacent busy and idle period) is

$$\mathrm{Var}(\mathrm{BC}) = \frac{\mu^2(1-\rho)^3 + \lambda^2(1+\rho)}{\lambda^2\mu^2(1-\rho)^3}.$$

Notation	Description
λ	The average arrival rate.
$A(t)$	The distribution of the interarrival process. For the M/M/1, $A(t) = 1 - e^{-\lambda t}$. The arrival process forms a renewal process.
μ	The average service rate.
$B(t)$	The distribution of the service time process. For the M/M/1 queue, $B(t) = 1 - e^{-\mu t}$.
m	The average service time. This notation is used for more general processes (with non-exponential service times). $m = \mu^{-1}$.
ρ	Traffic intensity or utilization. $\rho = \lambda\mu^{-1} = \lambda m$. Sometimes we denote ρ by U for utilization.
$Q(t)$	The queue length process as a function of time.
$N(t)$	The number in the system process as a function of time. As $t \to \infty$, $P(N \leq n) = 1 - \rho^{n+1}$.
$E(N)$	The expected value of the equilibrium queue length process. $E(N) = \lambda(\mu - \lambda)^{-1} = \rho(1-\rho)^{-1}$.
Var(N)	The variance of the equilibrium queue length distribution. Var(N) $= (\lambda\mu)(\mu - \lambda)^{-2} = \rho(1-\rho)^{-2}$.
$\mathcal{I}$	A random variable describing an idle interval of the equilibrium process. For the M/M/1, $\mathcal{I} \sim \exp(\lambda)$.
$E(\mathcal{I})$	The expected length of an idle interval, λ^{-1}.
$\mathcal{B}$	A random variable describing a busy period of the equilibrium process. For the M/M/1, the distribution of $\mathcal{B}$ is known. See [35].
$E(\mathcal{B})$	The expected value of a busy period duration. $E[\mathcal{B}] = (\mu - \lambda)^{-1}$.
N_b	The expected number served in a busy period. $N_b = \mu/(\mu - \lambda)$. The distribution is also known.
$R(t)$	The equilibrium response time distribution for a fifo queue (includes waiting plus service). $R(t) = 1 - e^{-(\mu-\lambda)t}$.
$E(R)$	The mean of the equilibrium response time distribution. $E(R) = (\mu - \lambda) = \mu^{-1}(1-\rho)^{-1}$.
Var(R)	The variance of the equilibrium respone time distribution. Var(R) $= (\mu - \lambda)^{-2} = \mu^{-2}(1-\rho)^{-2}$.
$W(t)$	The equilibrium waiting time distribution for a fifo server. $W(t) = 1 - \rho e^{-\mu(1-\rho)}$.
n_k	The probability the system is in state k, or equivalently, contains k tasks where $k \in \{0, 1, 2, ...\}$. $n_k = (1-\rho)\rho^k$.
$E[k \to k-1]$	The mean time to first transition from state k to state $k-1$, $k \in \{1, 2, 3, ...\}$. $E[k \to k-1] = (\mu - \lambda)^{-1}$.

Table B.1: M/M/1 Steady State Distribution Variables

Appendix C

Limit Theorems

C.1 CLT for Renewal Processes

Lemma C.1.1 (Central Limit Theorem for Renewal Processes.) *Let X_1, X_2, $X_3, \ldots$ be independent and identically distributed with $E[X_j] = \mu$ and $Var[X_j] = \sigma^2$. Define $S_j = \sum_{m=1}^{j} X_m$. Let $N(t)$ be the counting process for the sucessive times between the X_j. In symbols, $N(t) = \max\{j : S_j \leq t\}$.*

Then,

$$\lim_{t\to\infty} \Pr\{\frac{N(t) - t/\mu}{\sqrt{t\sigma^2\mu^{-3}}} < x\} = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-y^2/2}\, dy.$$

As an application of the CLT for renewal processes, consider an approximation for the number of idle intervals in an M/M/1 queue during a hyperperiod of length H. Then,

$$m = \frac{1}{\lambda} + \frac{1}{(\mu - \lambda)} = \frac{\mu}{\lambda(\mu - \lambda)},$$

$$\sigma^2 = \frac{1}{\lambda^2} + \frac{(1+\rho)}{\mu^2(1-\rho)^3} = \frac{\mu^2(1-\rho)^3 + \lambda^2(1+\rho)}{\lambda^2\mu^2(1-\rho)^3}$$

so to approximate N_H, the number of idle intervals in $[0, H)$ (or $[kH, (k+1)H)$ for $k \in \{0, 1, 2, \ldots\}$) when H is large we have

$$N(H) \sim \mathcal{N}(H\lambda(1-\rho), H\lambda(1 - 3\rho + 4\rho^2)).$$

C.2 The PASTA Property

Lemma C.2.1 (The PASTA Property.) *PASTA stands for "Poisson Arrivals See Time Averages". More technically, let* $S = \{S(t)|t \geq 0\}$ *define the system state. Let* B *be an arbitrary collection of states in* S. *Let* $P = \{P(t)|t \geq 0\}$ *be a Poisson process with rate* $\lambda > 0$. *In applications considered in this thesis,* P *will generally be the arrival process of aperiodic tasks to the system defined by* S, *but it need not be an arrival process at all.*

We say that S *is nonanticipating with respect to* P *if, for every* $t \geq 0$, $\{P(t+u) - P(t)|u \geq 0\}$ *is independent of both* $\{S(v)|0 \leq v \leq t\}$ *and* $\{P(v)|0 \leq v \leq t\}$.

Let $U(t) = [S(t) \in B]$, *that is* $U(t)$ *is the indicator function of* $S(t)$ *in the set* B. *Also, define*

$$\bar{U}(t) = (\int_0^t U(s)ds)/t,$$

$$A(t) = \int_0^t U(s)dP(s),$$

and

$$\bar{A}(t) = (\int_0^t U(s)dP(s)) = A(t)/P(t).$$

Then, when U *is nonanticipating with respect to* P *and has left-continuous sample paths with right limits,* $\bar{U}(t) \to \bar{U}(\infty)$ *w.p.1 if and only if* $\bar{A}(t) \to \bar{U}(\infty)$ *w.p.1, as* $t \to \infty$.

In practical terms, this theorem says that the fraction of Poisson events (arrivals) that find (see) $S \in B$ *(i.e.* $\bar{A}(t)$*) is the same as the time average of the system in state* B *(i.e.* $\bar{U}(t)$*).*

For a proof of the PASTA property, see [38].

An application of the PASTA property gives the virtual waiting time distribution for Poisson arrivals is equal to the actual waiting time distribution.

C.3 Brownian Motion

Let ξ_t be a Weiner process with density

$$f_t(x) = \frac{1}{\sqrt{2\pi t}} e^{-x^2/2t}.$$

Lemma C.3.1 (Brownian Motion with Reflection at Zero.) *Let B_t be a Brownian motion with drift $m < 0$, variance σ^2 and with a lower control barrier that behaves like reflection at the origin. For $B_t > 0$, we have $B_t \sim \mathcal{N}(mt, \sigma^2 t)$. Let $M_t = \sup_{0 \le s \le t} B_s$. It can be shown that $M_t \sim B_t$ [14].[1] Let $t \to \infty$, so $M_\infty \sim B_\infty$. It is well known ([2],[14],[20]) that the limiting distribution of M_t is given by*

$$\lim_{t \to \infty} \Pr(M_t \le x) \; 1 - e^{2mx/\sigma^2},$$

hence

$$\lim_{t \to \infty} \Pr(B_t \le x) \; 1 - e^{2mx/\sigma^2}.$$

Lemma C.3.2 (Linear Combinations of Weiner Processes.) *Let ξ_1, ξ_2 and ξ be Weiner processes with ξ_1 and ξ_2 independent. Then for constants c_1 and c_2,*

$$c_1\xi_1 + c_2\xi_2 \sim (c_1^2 + c_2^2)^{\frac{1}{2}} \xi.$$

The result is an easy exercise.

[1] *This is not obvious. See Section 1.9 of [14].*

C.4 Donsker's Theorem

Virtually all the work referenced in this thesis that makes use of weak convergence arguments to find limiting distributions are based on Donsker's Theorem. We have included a statement of a version of Donsker's theorem for the sake of completeness.

Lemma C.4.1 (Donsker's Theorem.) *Let* $\mathrm{D} = \mathrm{D}[0,1]$ *be the space of functions on* $[0,1]$ *with discontinuities of at most the first kind. Let* $Y_1, Y_2, \ldots$ *be iid random variables defined on* $(\Omega, \mathcal{B}, \mathbf{P})$ *with partial sums* $S_n = Y_1 + Y_2 + \ldots + Y_n$. *Further suppose,* $E[Y_n] = 0$ *and* $\mathrm{Var}(Y_n) = \sigma^2 < \infty$. *Define* X^n *at each time* t *by*

$$X^n(t,\omega) = \frac{1}{\sigma\sqrt{n}} S_{[nt]}(\omega).$$

Note that for each $\omega \in \Omega, X^n(\omega) \in \mathrm{D}$. *Then* $\lim_{n\to\infty} X^n(t) \Rightarrow \xi_t$ *where* ξ_t *is a Weiner process, or equivalently* $\xi_t \sim \mathcal{N}(0,t)$.

Proofs of Donsker's theorem can be found in [4], Theorems 10 (pg 68) and 16.1 (pg 137). Donsker's theorem is also known as the invariance principle *and as the* functional central limit theorem.

Many extensions to Donsker's theorem have been made. For example, specification of non-zero means, relaxations of the *iid* assumption for $Y_1, Y_2, \ldots$ and increases in the domain of D (e.g. $\mathrm{D}[0,\infty)$). These modified versions have been applied and/or developed to obtain weak convergence results for queueing systems in heavy traffic.

In all cases (we used), a sequence of scaled processes is defined where time is scaled by n^{-1} and space (e.g. queue size) is scaled by $n^{-1/2}$ in the n^{th} process. We denote the n^{th} (scaled) process at time t by $X_t^n = X^n(t)$. As $n \to \infty$ the limiting diffusion is denoted by $X^\infty(t) = X(t) \sim \xi_t$. The scaled sequence is superscripted leaving subscripts for task class.

Examples using these scalings applied to queues can be found in [5], [13], [15], [16], [18], [25], [26] and [37], to cite just a few. The limiting process (as $n \to \infty$) defines the diffusion process for which we seek an equilibrium solution (when it exists), subject to suitable boundary conditions. Chapters 2 and 3 contain more detailed examples illustrating scaled processes and applications of weak convergence arguments based on Lemma C.4.1. Alternative scalings might be useful for lighter loadings.

Appendix D

Algorithms and Computations

D.1 Some Model and Parameter Calculations

D.1.1 Long Hyperperiod Models

In Section 5.2.2, a response time CDF for the background aperiodic scheduling discipline was developed for long hyperperiods. In that development, a point mass at x_0 defined the probability that a newly arriving task experienced no blocking delays incurred by periodic tasks. In our system, the actual response time distribution will be continuous. For the BGA Long Hyperperiod Model (LHM) computation, x_0 can be thought of as the point at which the response time behavior for values of $r < x_0$ is defined by an M/M/1 queue, and for values $r > x_0$ response times are defined by the LHM developed in Section 5.2.2. For this definition, x_0 satisfies

$$R_m(x_0) = \omega_0 = 1 - e^{-(\mu_2 - \lambda_2)x_0} = R_l(x_0) = 1 - \rho_1(1 - \rho_2)^{-1}$$

with a simple calculation giving

$$x_0 = \frac{-\ln((\rho_1)(1 - \rho_2)^{-1})}{(\mu_2 - \lambda_2)}.$$

x_0 decreases for fixed ρ_1 and increasing ρ, which one would expect since the response time range for which the M/M/1 applies will decrease as total traffic increases. Also, for fixed ρ, x_0 increases with decreasing ρ_1 which would also be expected since the M/M/1 portion of the interval increases as ρ_1 decreases.

Using this definition for x_0, the resulting response time CDF is given in Equation D.1.

$$R_l(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 - e^{-(\mu_2 - \lambda_2)x} & \text{for } 0 \leq x \leq x_0 \\ 1 - \frac{\rho_1}{(1-\rho_2)}(1 - \frac{(x-x_0)}{B}) & \text{for } x_0 \leq x \leq B + x_0 \\ 1 & \text{for } x \geq B + x_0 \end{cases} \tag{D.1}$$

Except for very long hyperperiods our data suggests that this value of x_0 errors on the side of being too small, especially when the system is moderately loaded.

When deriving the LHM for aperiodic system size in Section 5.3.2, we assumed a value for n_0, the aperiodic system size at the start of the hyperperiod. For consistency, we define n_0 as the aperiodic workload (x_0) divided by the mean processing time per aperiodic task (μ_2^{-1}), or equivalently, $n_0 = x_0\mu_2$, which is also written as

$$n_0 = \frac{-\ln((\rho_1)(1-\rho_2))^{-1}}{(1-\rho_2)}.$$

Table D.1 lists several values of n_0 for varying ρ_2 and fixed ρ_1. Like x_0, n_0 decreases for fixed ρ_1 and increasing ρ, which one would expect since the response time range for which the M/M/1 applies will decrease as total traffic increases. Also, for fixed ρ, n_0 increases with decreasing ρ_1 which would also be expected since the M/M/1 portion of the interval increases as ρ_1 decreases.

To calculate x_0 (and n_0) for the foreground aperiodic scheduling policy, one must first observe from the data either $\bar{B}$, or ω_0 where

$$\tilde{\omega}_0 = \omega_0 + (1 - \omega_0)(1 - \frac{1}{\beta H(1-\rho_2)}) = 1 - \frac{\bar{B}}{H(1-\rho_2)},$$

and β is the mean blocking rate given some blocking occurs. Recall that in the FGA

ρ	ρ_2	ρ_1	N_2	n_0
0.99	0.74	0.25	2.846	0.0523
0.85	0.60		1.500	1.1750
0.80	0.55		1.222	1.3060
0.99	0.24	0.75	0.316	0.0174
0.85	0.10		0.111	0.2030
0.80	0.05		0.053	0.2490

Table D.1: Values for n_0 in the LHM

LHM, maximum blocking does not occur, and β can be derived using either heavy traffic theory or the DSM parameter.

D.1.2 Response Times for Background Aperiodics

Refer back to Section 5.2.4 for process and algorithm definitions. This is the (process) state variable simulation algorithm used to collect the region percentiles for the PWLM when estimating response times under the background aperiodic scheduling discipline. The algorithm was devised for intermediate hyperperiods, but also works reasonably well for short and long hyperperiods.

```
-- function LPWM Estimation(H,C : in time) return (V, P*) : CDF;
-- This is process state simulation code to construct a CDF for the
-- BGA discipline; interemediate hyperperiod case.
 -- constant declarations
 hcmin : counter := 500; acmin : counter := 1500;
 minp : time := min(C, H − C); maxp : time := max(C, H − C);
 -- variable declarations
 hc,ac: counter := 0;
 tcurr,lcurr,tlmodH,tcmodH : time := 0;
 tcdivH, tldivH, k : natural := 0;
 iarr, svc : (random exponential) time := 0;
 Wa, vmax : time := 0;
 i1,i2, vl : index := 1;
begin
 -- the hyperperiod count loop
 while (hc < hcmin or ac < acmin) loop
 hc := hc +1;
  -- the aperiodic task count loop
  while (tcdivH ≤ tldivH) loop
   ac := ac +1;
   iarr := exp(interarrival time); svc := exp(service time);
   Wa := max(0,Wa - (tcurr - tlast)) + svc;
   if Wa > vmax then vmax := Wa; end if;
   k := Wa div (H-C);
   tlast := tcurr; tlmodH := tcmodH; tldivH := tcdivH;
   tcurr := tcurr + iarr; tcdivH := tcurr div H; tcmodH := tcurr mod H;
   if (((tcmodH ≤ minp) and (Wa ≤ (k − 1)(H − C)+ tcmodH)) or
     ((minp ≤ tcmodH ≤ maxp) and (Wa ≤ (k − 1)(H − C)+ minp)) or
     ((maxp ≤ tcmodH) and (Wa ≤ (k − 1)(H − C)+ tcmodH - minp))) then
    i1 := 2k - 1;
   else i1 := 2k;
   end if;
   P(i1) := P(i1) + 1; if vmax = Wa then vl := i1; end if;
  end while;
 while;
 -- P = P/ac; Now calculate the return value (V, P*) using the steps in Section 5.2.3
 return (P*, V);
end LPWM Estimation;
```

Figure D.1: BGA IHM LPWM Estimation for Response Times Distributions

Bibliography

[1] Soren Asmussen, "Applied Probability and Queues", John Wiley & Sons, 1987

[2] Ludwig Arnold, "Stochastic Differential Equations: Theory and Applications", Krieger Publishing Company, 1992 (Reprint Edition); Original English Translation Edition 1974 (John Wiley & Sons, Inc.)

[3] Peter J. Bickel and Kjell A. Doksom, "Mathematical Statistics: Basic Ideas and Selected Topics", Holden Day, 1977

[4] Patrick Billingsley, "Convergence of Probability Measures", J. Wiley & Sons, 1968

[5] David Y. Burman, "An Analytic Approach to Diffusion Approximations in Queueing," PhD Dissertation, Mathematics Department, New York University, February 1979

[6] H. Chetto and M. Chetto, "Some Results of the Earliest Deadline Scheduling Algorithm", *IEEE Transactions on Software Engineering*, Vol. 15, No. 10, Oct. 1989, pg. 1261-1269

[7] David M. DeLong, "Crossing Probabilities for a Square Root Boundary by a Bessel Process", *Commun. Statis.-Theor.Meth.*, A10(21), 1981, pg. 2197-2231

[8] Jean-Dominique Deuschel and Daniel W. Stroock, "Large Deviations," Academic Press, 1984

[9] M. D. Donsker, "Justification and Extension of Doob's Heuristic Approach to the Kolmogorov-Smirnov Limit Theorems," *Ann. Math. Stat.*, 23, pg. 277-281, 1952

[10] J. L. Doob, "Heuristic Approach to the Kolmogorov-Smirnov Theorems," *Ann. Math. Stat.*, 20, pg. 393-403, 1949

[11] Donald Gross and Carl M. Harris, "Fundamentals of Queueing Theory", Third Edition, John Wiley & Sons, 1998

[12] Marion G. Harmon and T.P. Baker, "An Ada Implementation of Marsaglia's Universal Random Number Generator", *Ada Letters*, Volume VIII, Number 2, March/April 1988

[13] M. J. Harrison, "Brownian models of queueing networks with heterogeneous customer populations", *Proceedings of the IMA Workshop on Stochastic Differential Systems*, Springer-Verlag, 1988, pp. 147-186

[14] M. J. Harrison, "Brownian Motion and Stochstic Flow Systems," reprint Robert E. Krieger Publishing Company, Inc., 1990; original John Wiley and Sons, Inc., 1985

[15] Donald L. Inglehart and Ward Whitt, "Multiple Channel Queues in Heavy Traffic.I", *Adv. Appl. Prob.*, **2**, 150-177-369, 1970(a)

[16] Donald L. Inglehart and Ward Whitt, "Multiple Channel Queues in Heavy Traffic.II: Sequences, Networks, and Batches", *Adv. Appl. Prob.*, **2**, 355-369, 1970(b)

[17] N.K. Jaiswal, "Priority Queues," Academic Press, 1968

[18] Daniel P. Johnson, "Diffusion Approximation for Optimal Filtering of Jump Processes and for Queueing Networks", Ph.D. Thesis, Department of Mathematics, University of Wisconson, Madison, 1983

[19] Samuel Karlin and Howard M. Taylor, "A First Course in Stochastic Processes", Academic Press, 1975

[20] Samuel Karlin and Howard M. Taylor, "A Second Course in Stochastic Processes", Academic Press, 1981

[21] Leonard Kleinrock, "Queueing Systems, Volume 1: Theory", John Wiley & Sons, 1975

[22] Leonard Kleinrock, "Queueing Systems, Volume 2: Computer Applications", John Wiley & Sons, 1976

[23] John P. Lehoczky, Lui Sha, and Ye Ding, "The Rate Monotonic Scheduling Algorithm: Exact Characterization and Average Case Behavior", *Proceedings of the* 10^{th} *Real-Time Systems Symposium, IEEE*, December, 1989, pp. 166-171

[24] John P. Lehoczky and Sandra Ramos-Thuel, "An Optimal Algorithm for Scheduling Soft-Aperiodic Tasks in Fixed-Priority Preemptive Systems", *Real-Time Systems Symposium, IEEE Proceedings*, December 1992

[25] John P. Lehoczky, "Real-Time Queueing Theory", *Real-Time Systems Symposium, IEEE Proceedings*, December 1996

[26] John P. Lehoczky, "Using Real-Time Queueing Theory to Control Lateness", *ACM Sigmetrics Conference Proceedings*, June 1997

[27] C.L. Lui and J. W. Leyland, "Scheduling Algorithms for Multiprogramming in a Hard Real Time Environment", *Journal of the ACM* 20(1), January 1973, pp 46-61

[28] G. Marsaglia, A. Zaman, W.W. Tsang, "Toward a Universal Random Number Generator", *Statistics and Probability Letters*, Volume 9, Number 1, pg 35-39, 1990

[29] C. Park and F.J. Schuurmann, "Evaluations of Barrier-Crossing Probabilities of Wiener Paths", *Journal of Appl. Prob.*, 1976, pg. 267-275

[30] Shivendra S. Panwar, Don Towsley, Jack K. Wolf, "Optimal Scheduling Policies for a Class of Queues with Customer Deadlines to the Beginning of Service", *Journal of the Association for Computing Machinery*, October 1988, Vol 35, No 4, pp. 832-844

[31] N.U. Prabhu, "Foundations of Queueing Theory", Kluwer Academic Publishers, 1997

[32] Sheldon M. Ross, "Introduction to Probability Models", Academic Press, Sixth Edition, 1997

[33] David Siegmund, "Boundary Crossing Probabilities and Statistical Applications", *Annals of Statistics*, Vol. 14, No. 2, 1986, pg 361-404

[34] Robert Serfling, "Approximation Theorems of Mathematical Statistics", John Wiley & Sons, 1980

[35] Lojos Takacs, "Introduction to the Theory of Queues", Oxford University Press, 1962

[36] Hideaki Takagi, "Queueing Analysis: A Foundation of Performance Evaluation", Volume 1: Vacation and Priority Systems, Part 1, North-Holland, 1991

[37] Ward Whitt, "Weak Convergence Theorems for Priority Queues: Preemptive-Resume Discipline", *Journal of Applied Probability*, Vol. 8, pp. 74-94, 1971

[38] Ronald W. Wolff, "Stochastic Modeling and the Theory of Queues", Prentice Hall, 1989